Markt&Technik

BOOK-WARE

Amiga 3D-Sprinter

Interaktive Echtzeit-Animation

Erzeugung und Bewegung von brillanten, spiegelnden Drahtgitter-, Plotter-, Schattierungs- und Schlagschattengrafiken.



Auf Diskette enthalten:
Die ablauffähige Version des Programms »3D-Sprinter« mit vielen Beispieldateien.
Dazu eine ausführliche Programmanleitung im Buch.

Amiga 3D-Sprinter

BOOK-WARE

Interaktive Echtzeit-Animation

Erzeugung und Bewegung von brillanten, spiegelnden Drahtgitter-, Plotter-, Schattierungs- und Schlagschattengrafiken

Markt&Technik Verlag AG

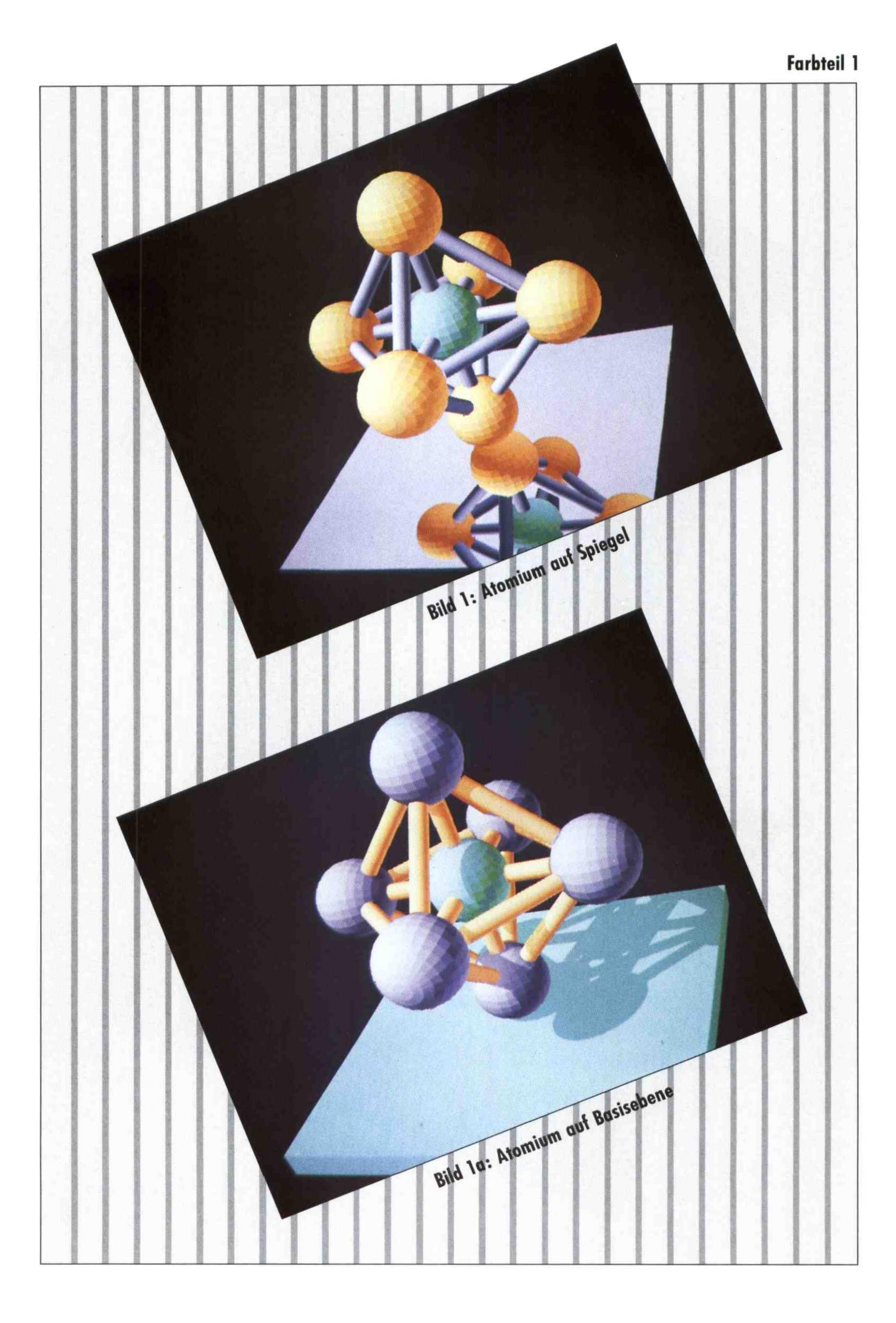


Bild 1: Atomium auf Spiegel

Bild 1a: Atomium auf Basisebene

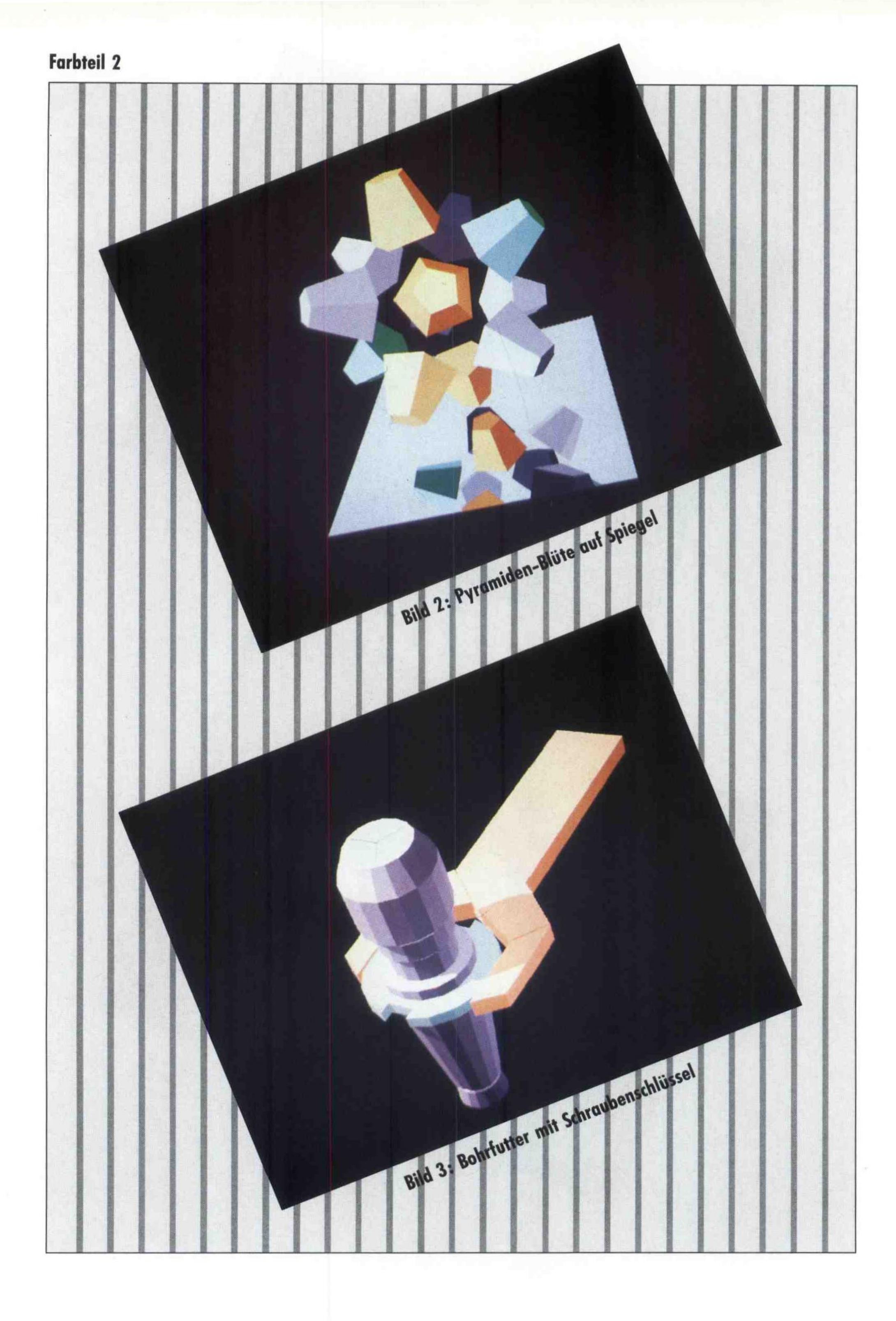


Bild 2: Pyramiden-Blüte auf Spiegel

Bild 3: Bohrfutter mit Schraubenschlüssel

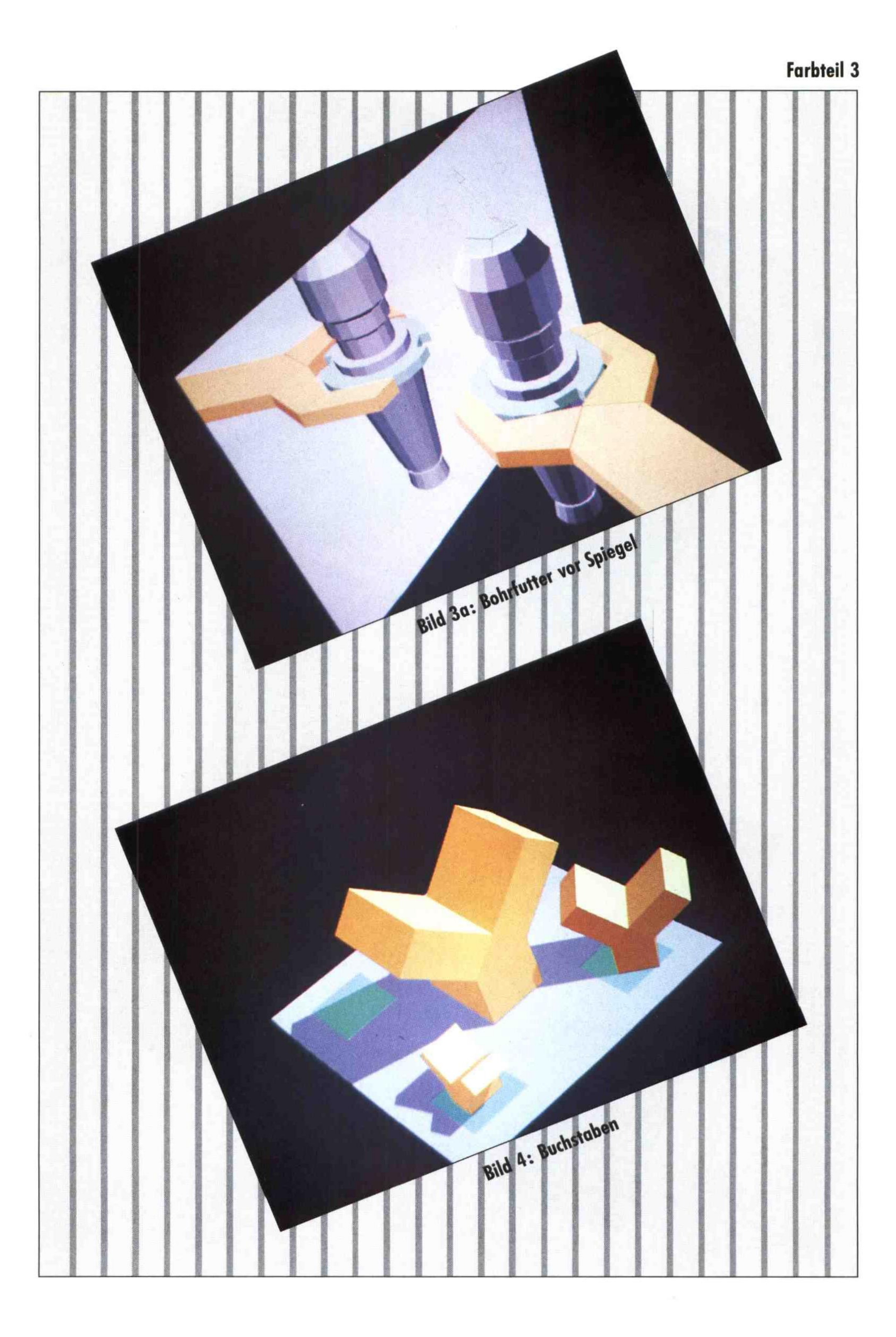


Bild 3a: Bohrfutter vor Spiegel

Bild 4: Buchstaben

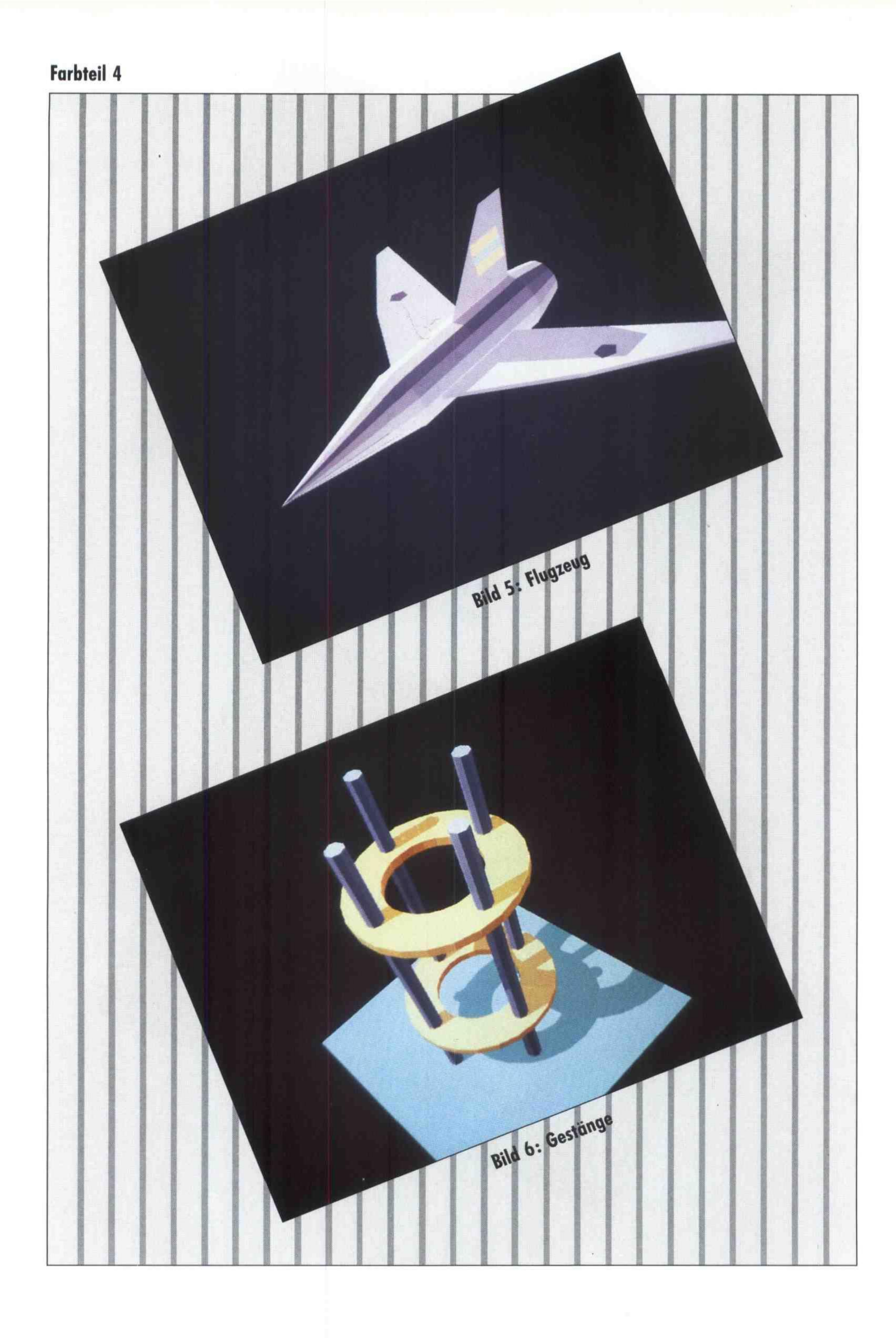


Bild 5: Flugzeug

Bild 6: Gestänge

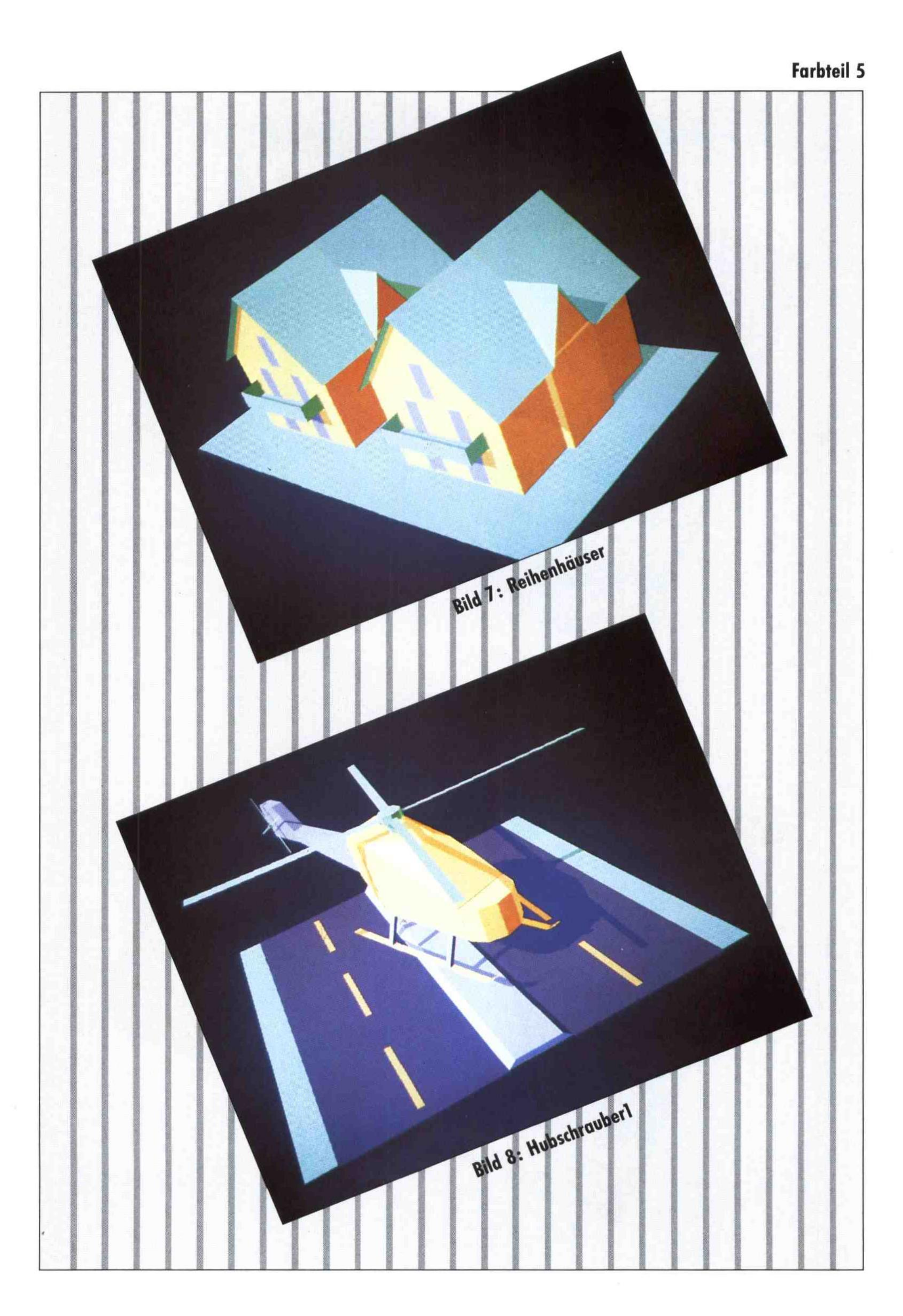


Bild 7: Reihenhäuser

Bild 8: Hubschrauber1

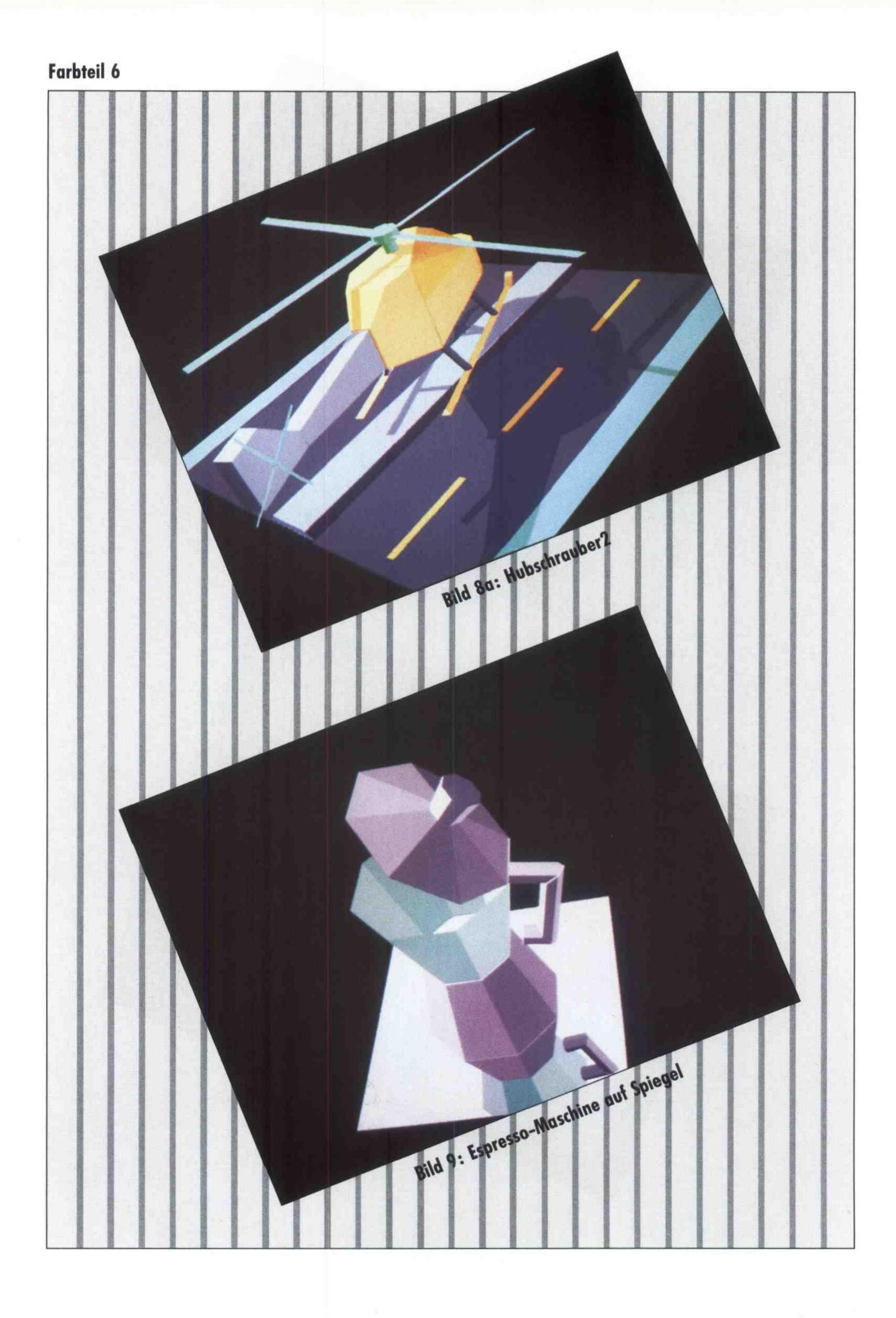


Bild 8a: Hubschrauber2

Bild 9: Espresso-Maschine auf Spiegel

Bild 9a: Espresso-Maschine

Bild 10: Dorfkirche

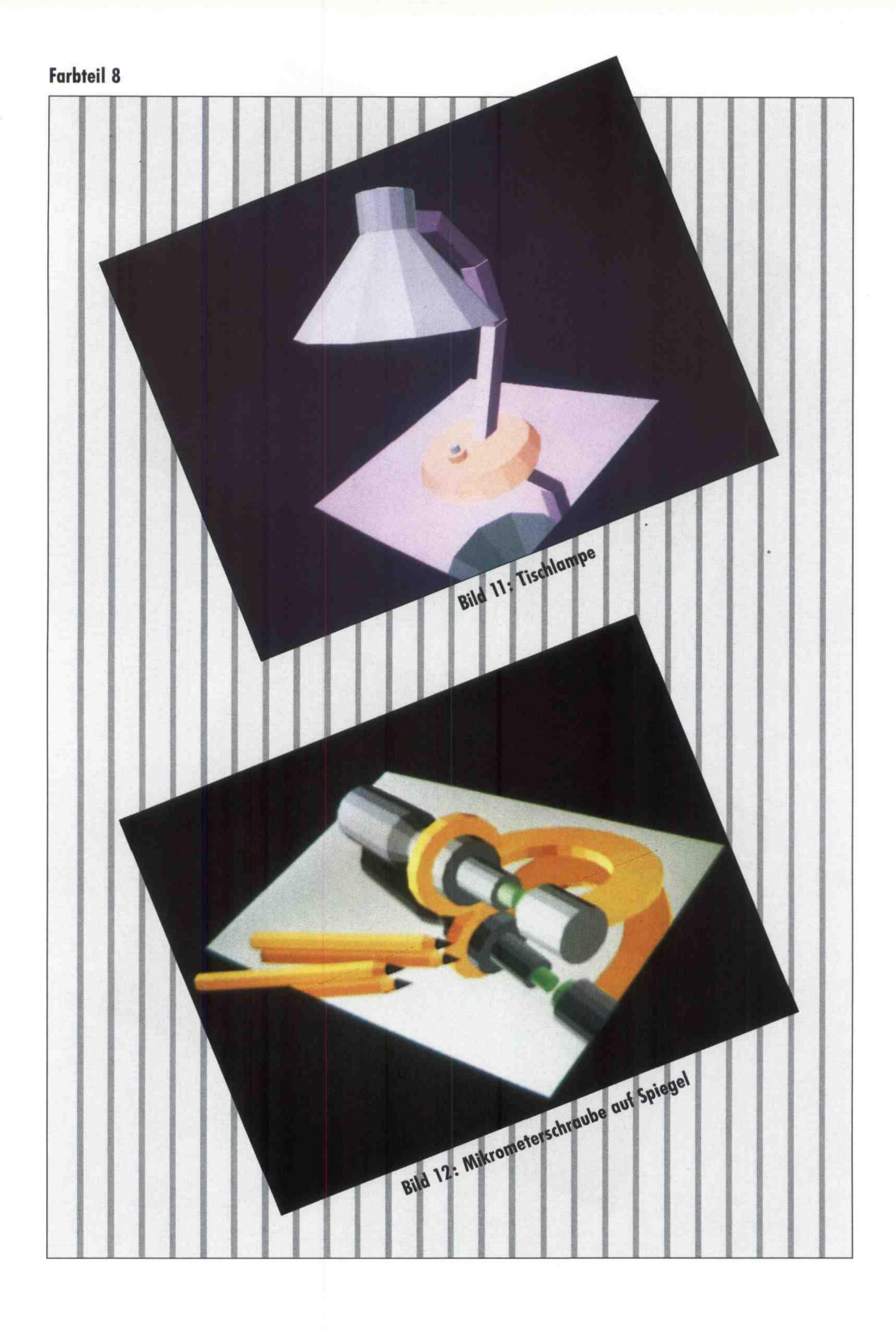


Bild 11: Tischlampe

Bild 12: Mikrometerschraube auf Spiegel

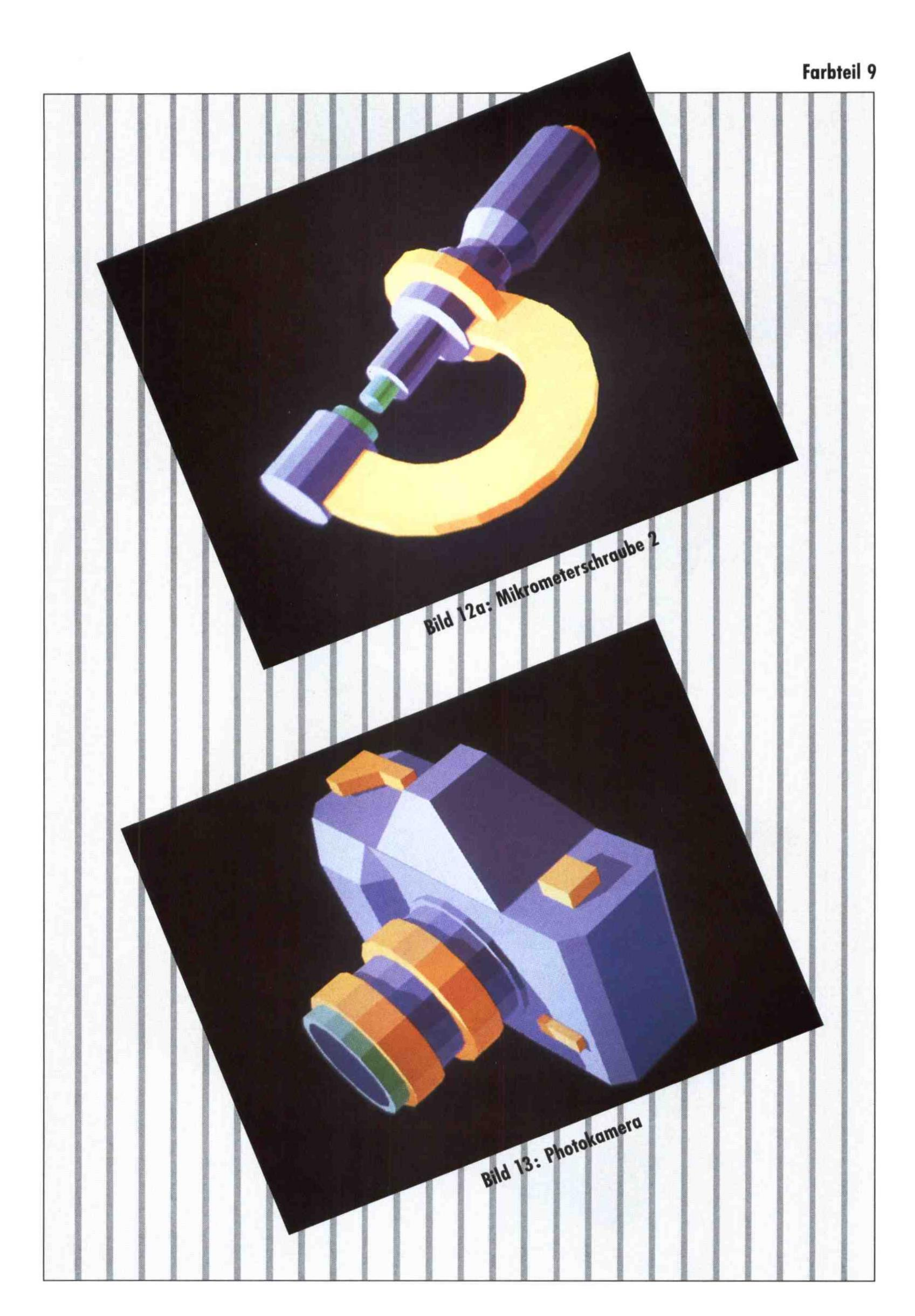


Bild 12a: Mikrometerschraube 2

Bild 13: Photokamera

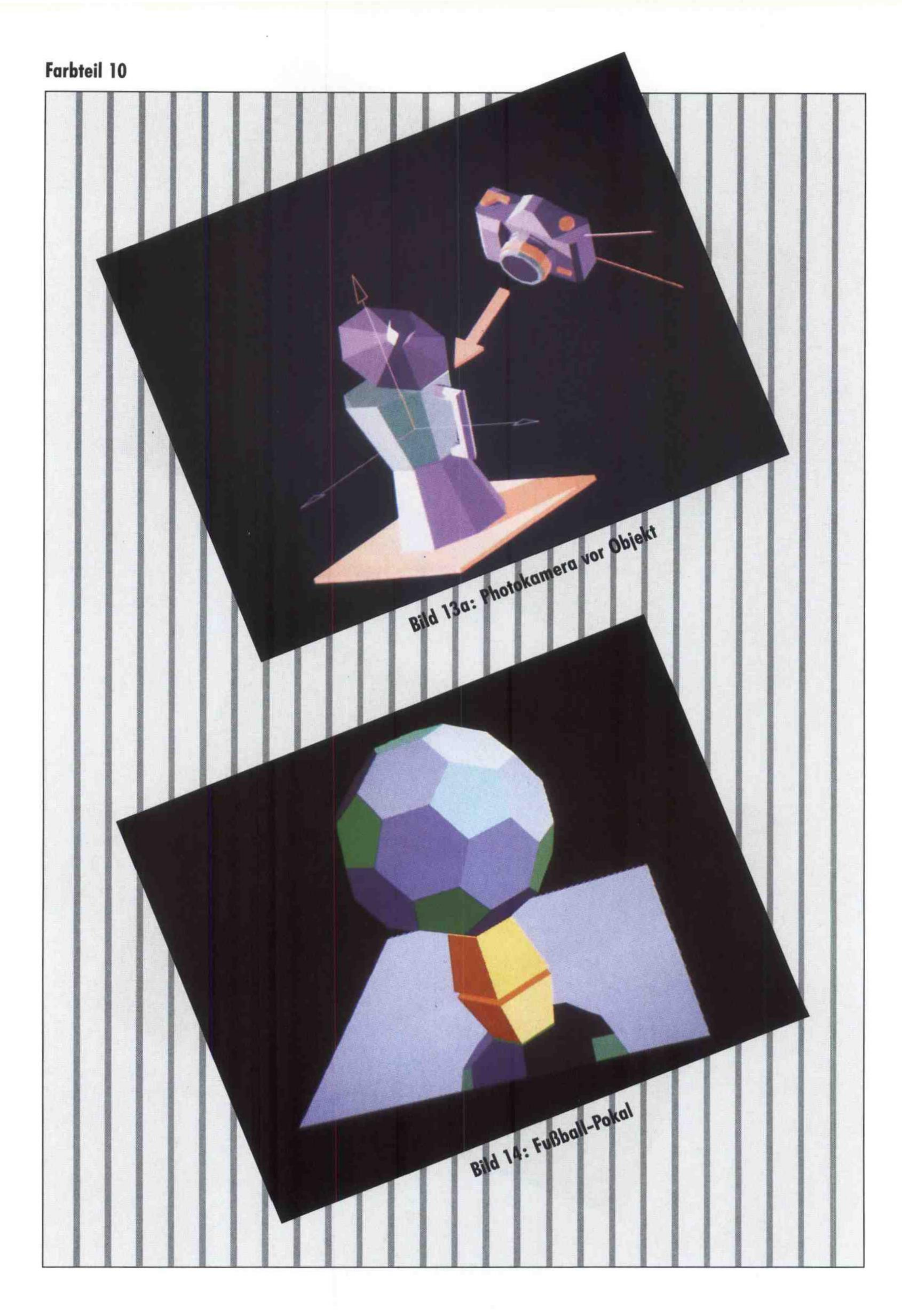


Bild 13a: Photokamera vor Objekt

Bild 14: Fußball-Pokal

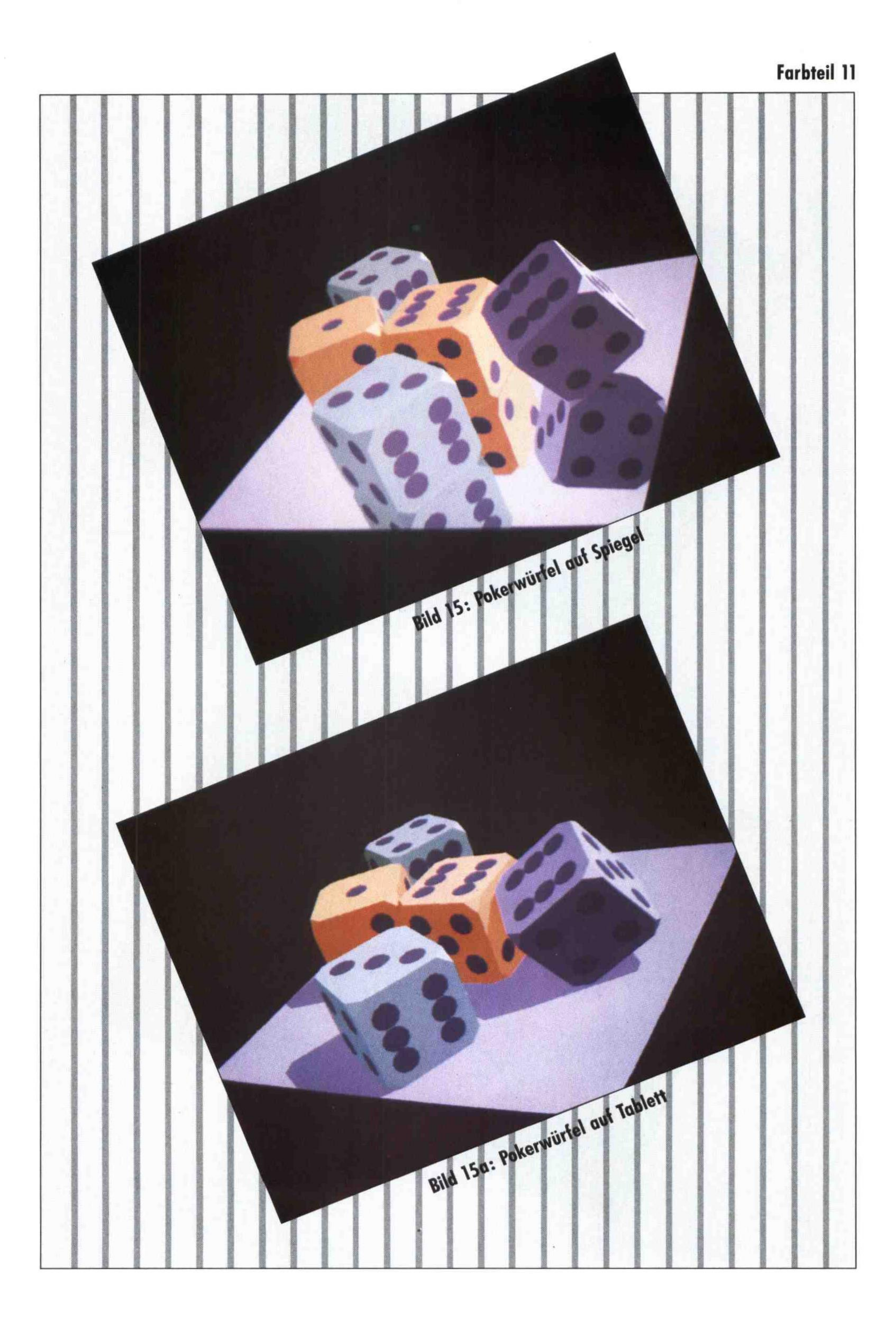


Bild 15: Pokerwürfel auf Spiegel

Bild 15a: Pokerwürfel auf Tablett

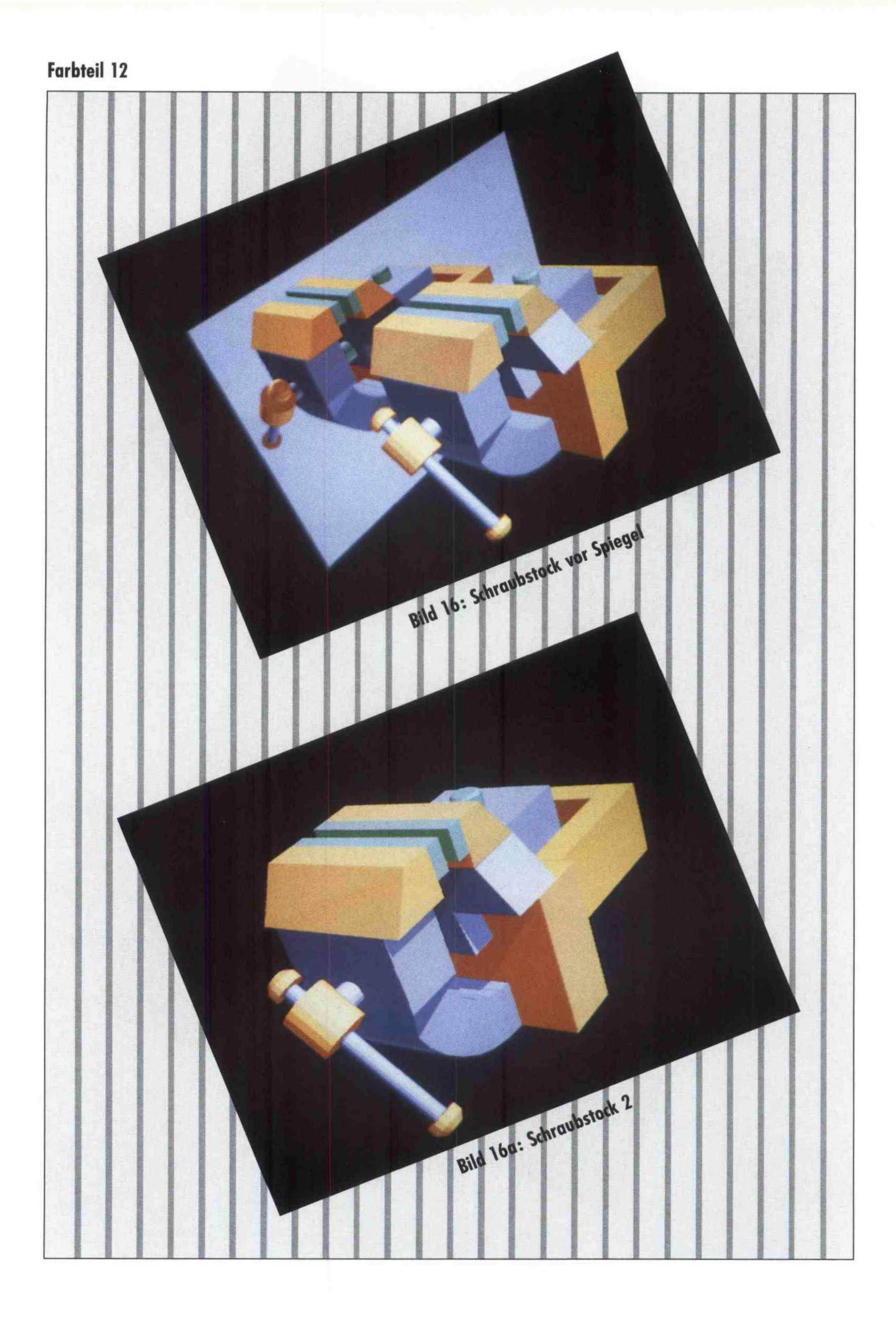


Bild 16: Schraubstock vor Spiegel

Bild 16a: Schraubstock 2

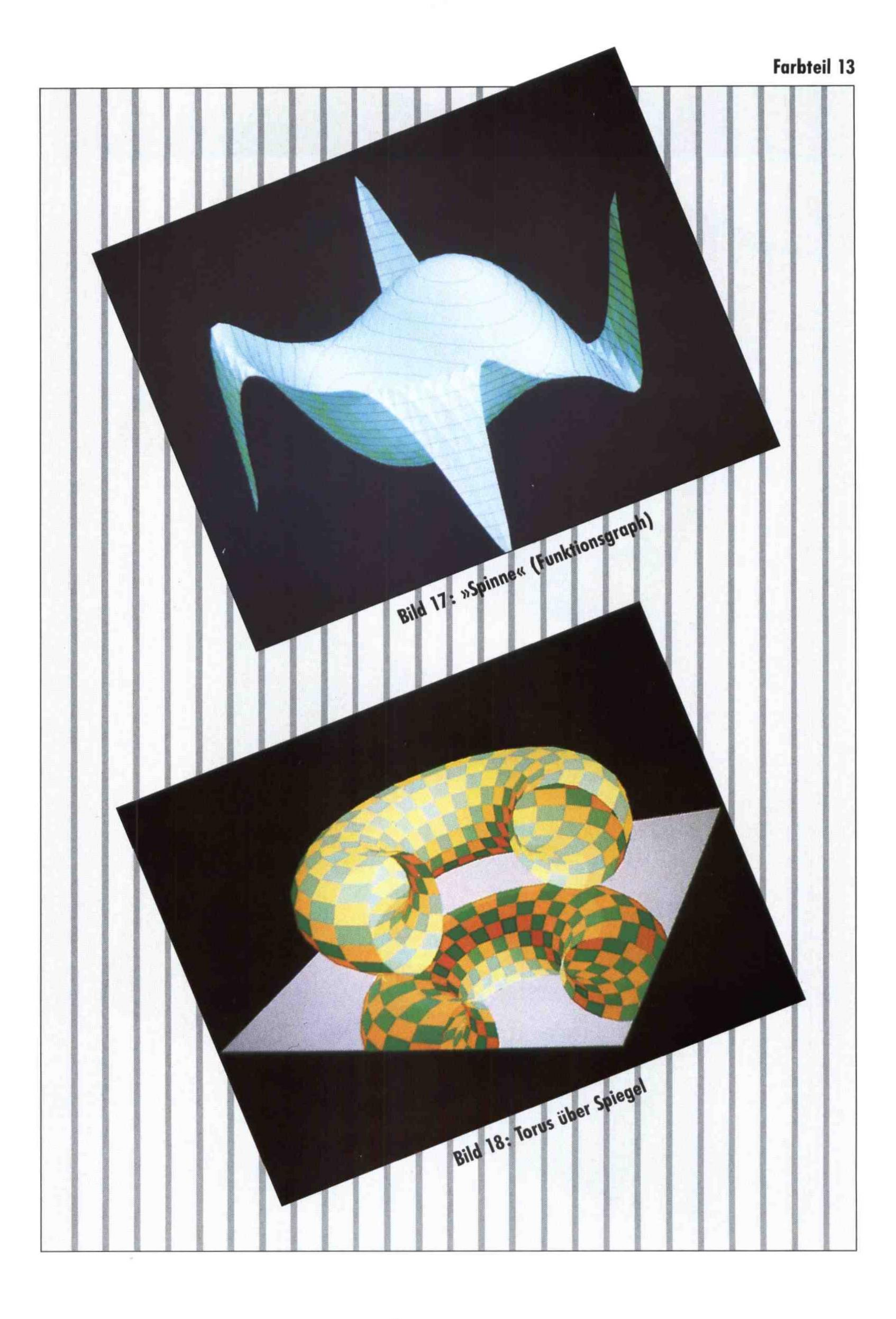


Bild 17: »Spinne« (Funktionsgraph)

Bild 18: Torus über Spiegel

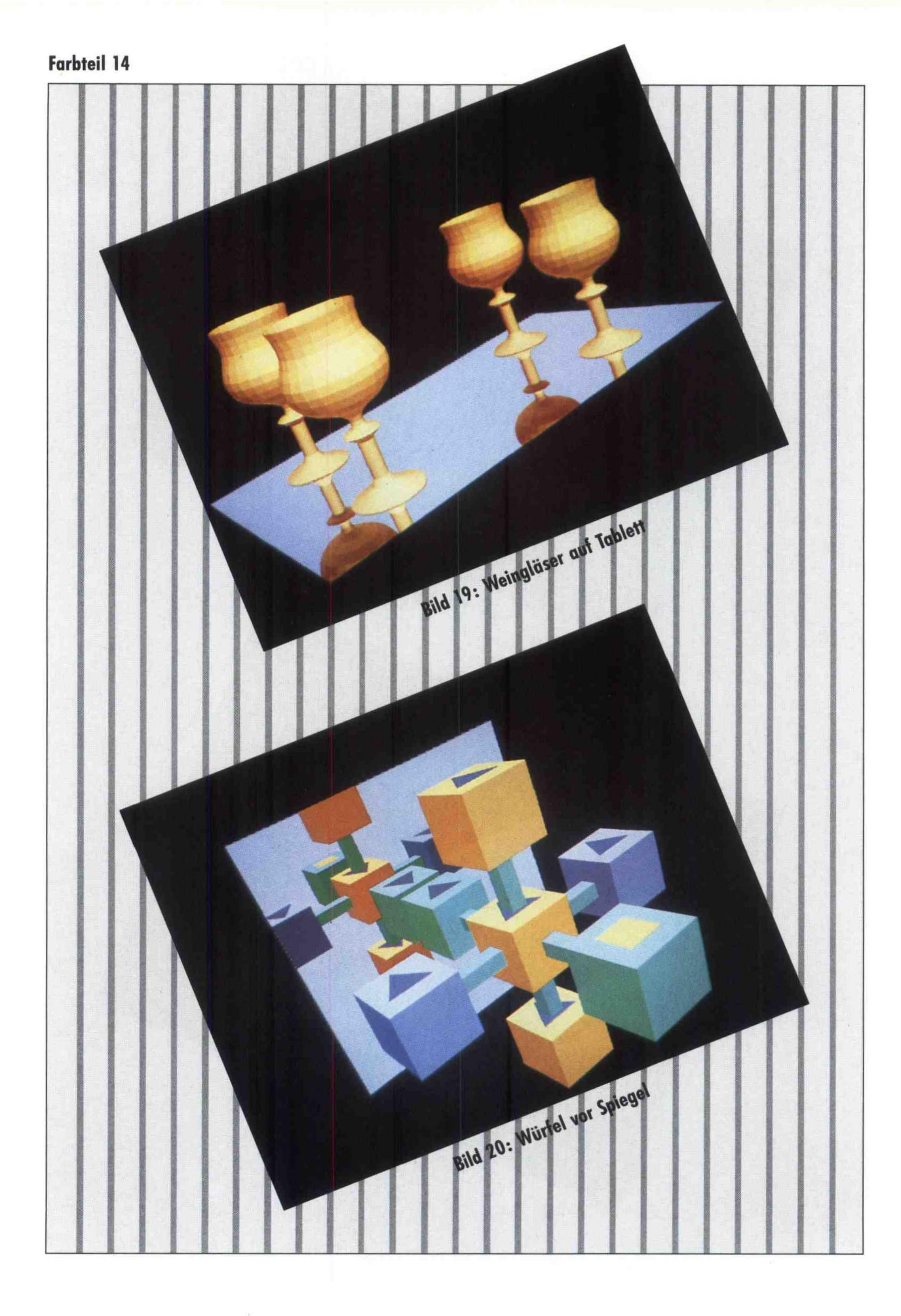


Bild 19: Weingläser auf Tablett

Bild 20: Würfel vor Spiegel

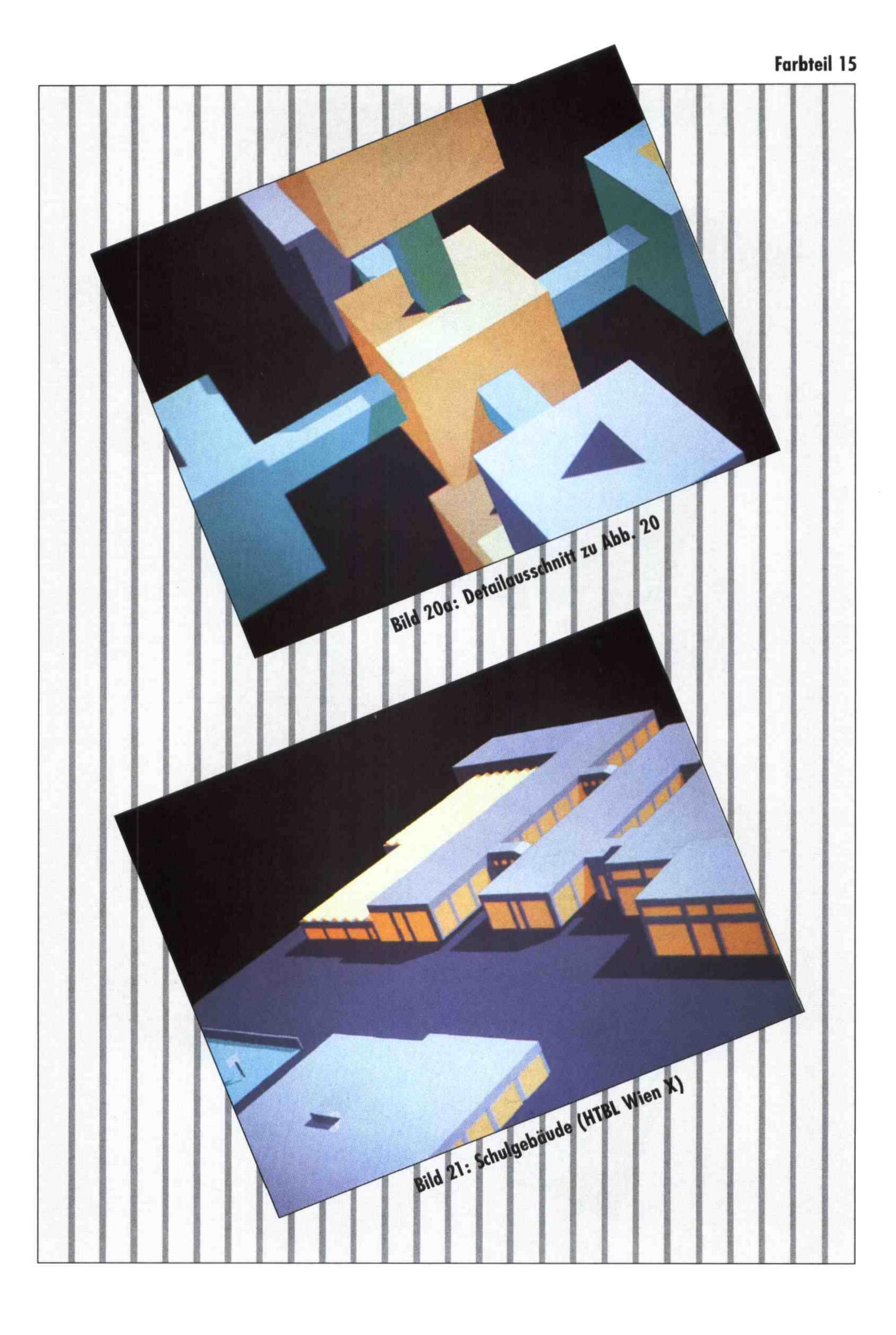


Bild 20a: Detailausschnitt zu Abb. 20

Bild 21: Schulgebäude (HTBL Wien X)

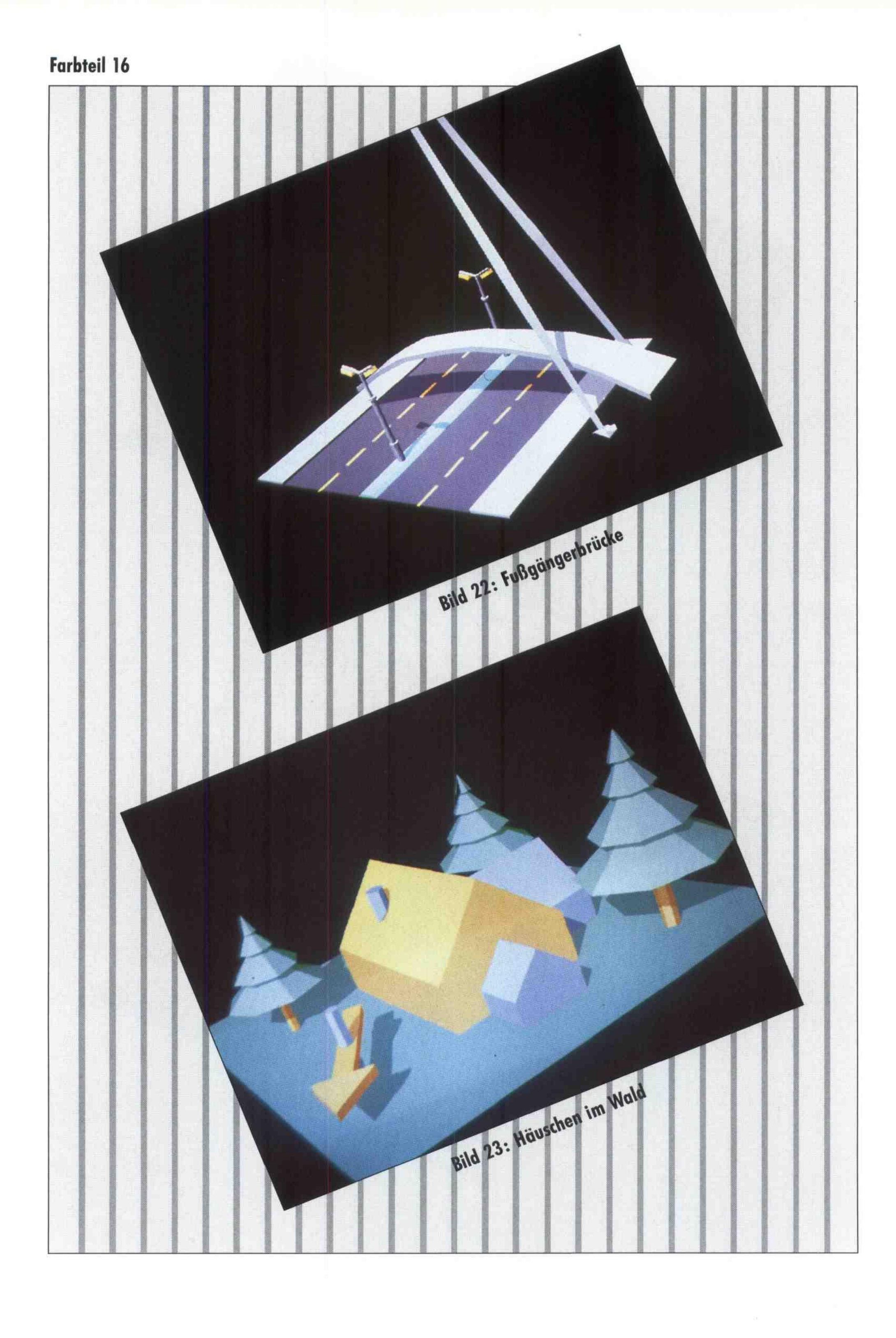


Bild 22: Fußgängerbrücke

Bild 23: Häuschen im Wald

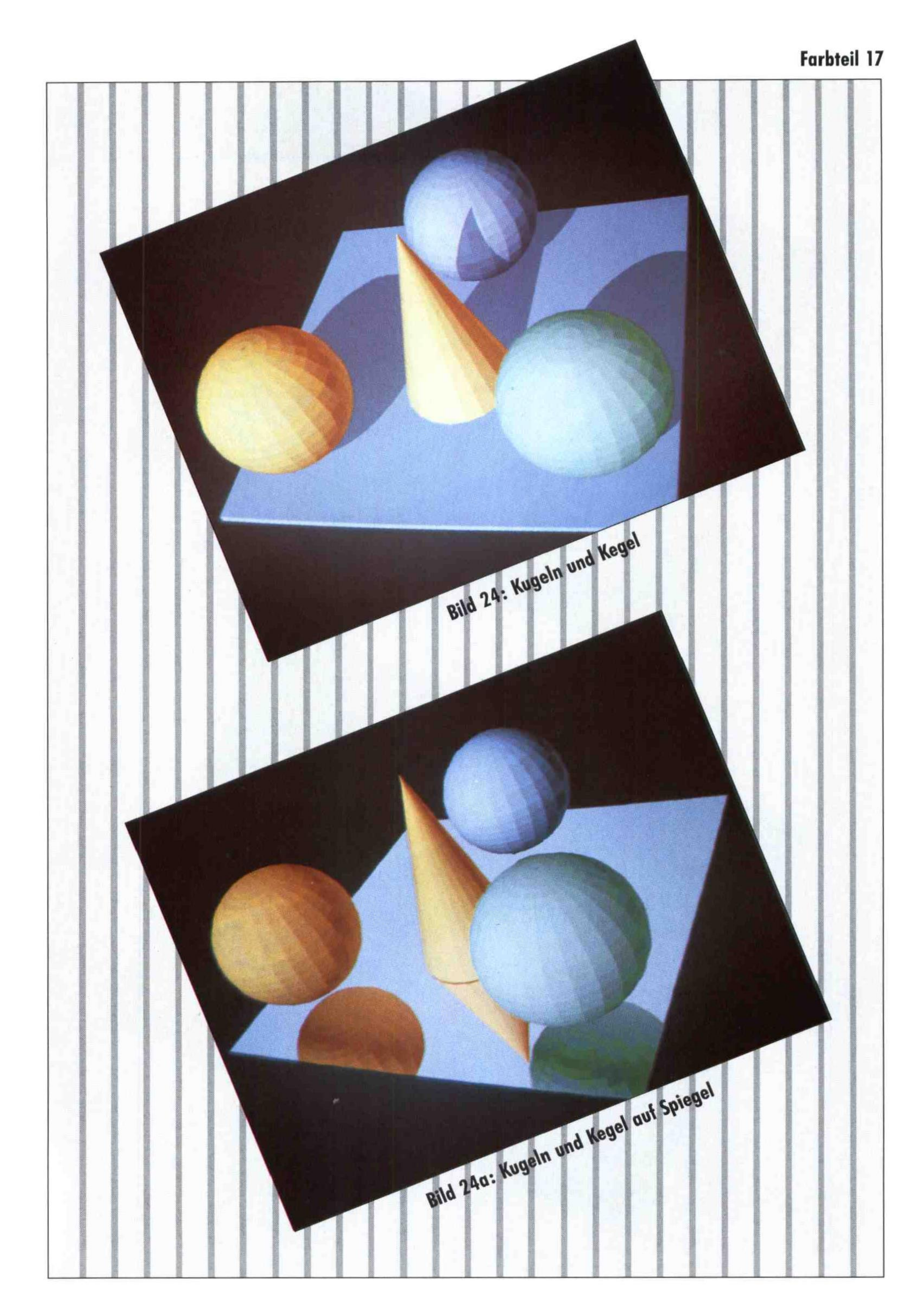


Bild 24: Kugeln und Kegel

Bild 24a: Kugeln und Kegel auf Spiegel

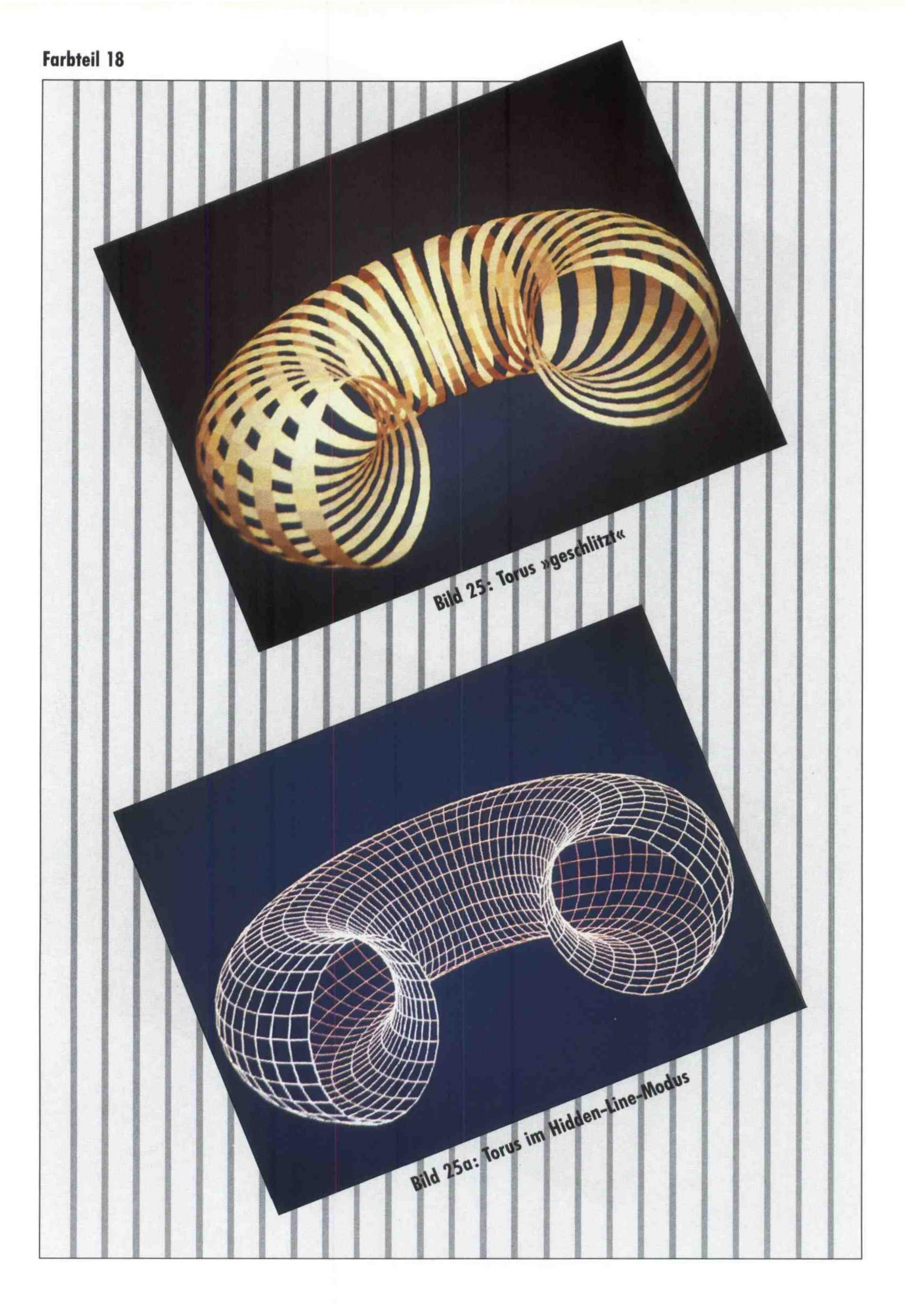


Bild 25: Torus »geschlitzt«

Bild 25a: Torus im Hidden-Line-Modus

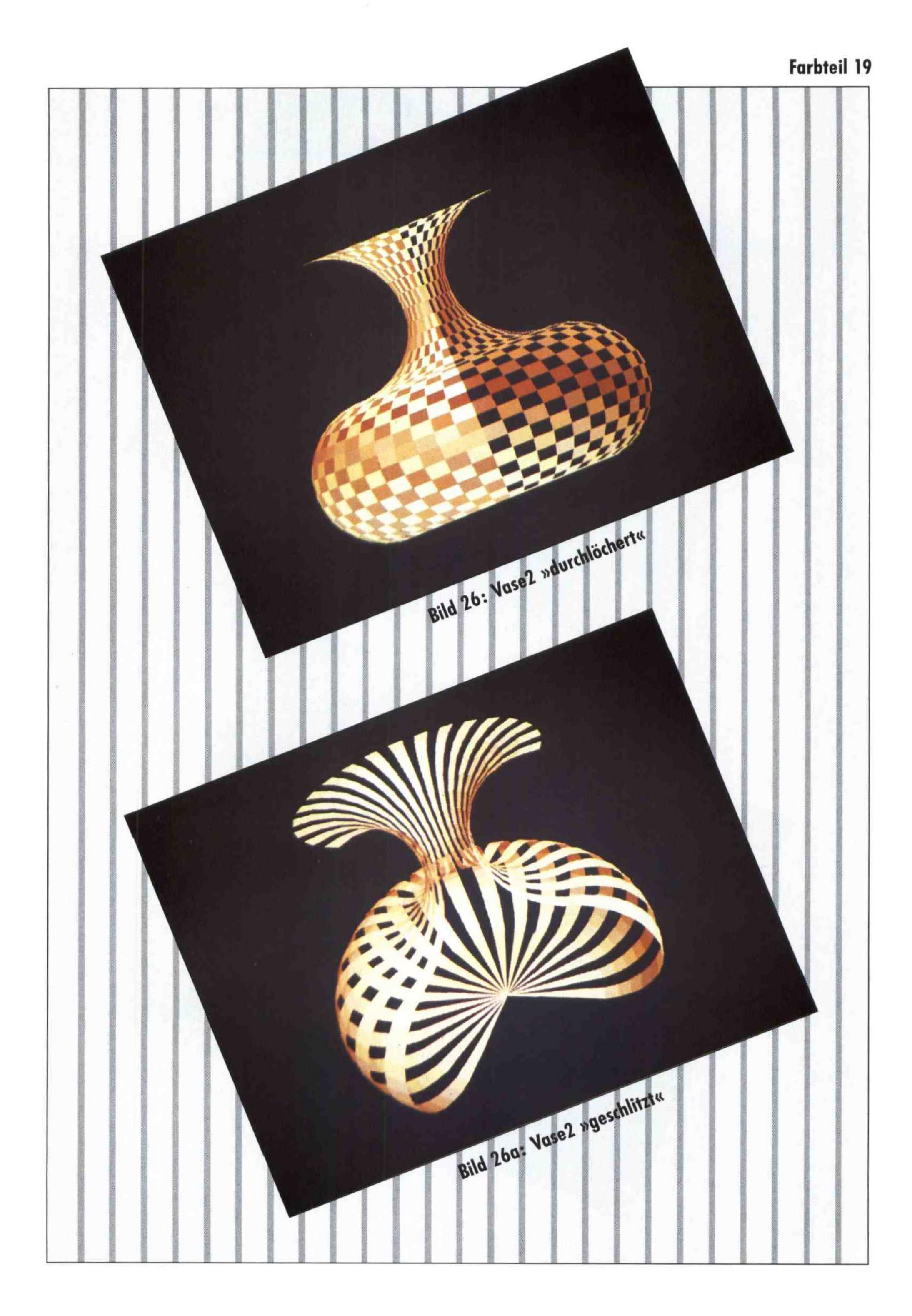


Bild 26: Vase2 »durchlöchert«

Bild 26a: Vase2 »geschlitzt«

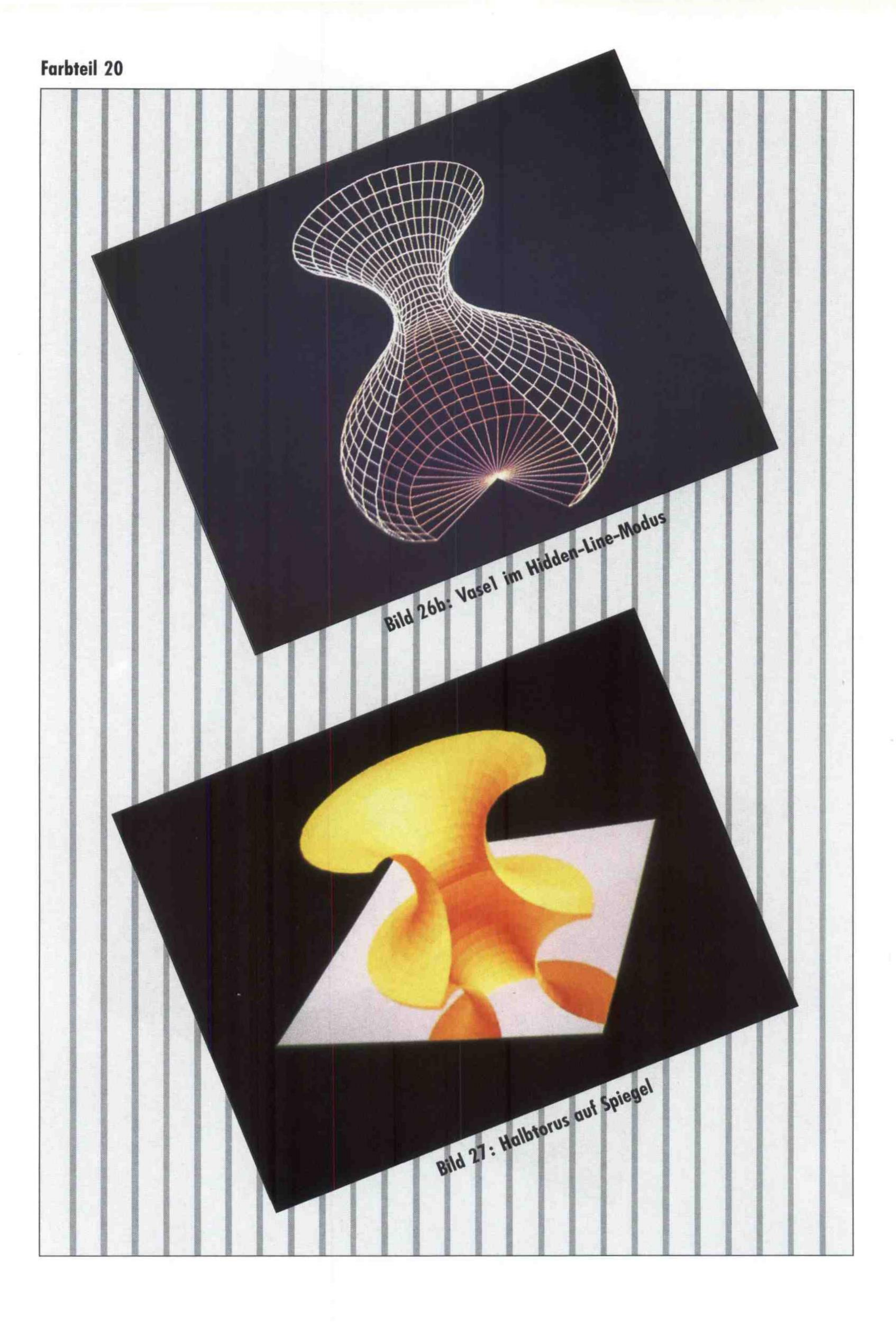


Bild 26b: Vasel im Hidden-Line-Modus

Bild 27: Halbtorus auf Spiegel

Bild 27a: Halbtorus »geschlitzt«

Bild 28: Geländefläche (berge.data) mit Schichtenlinien

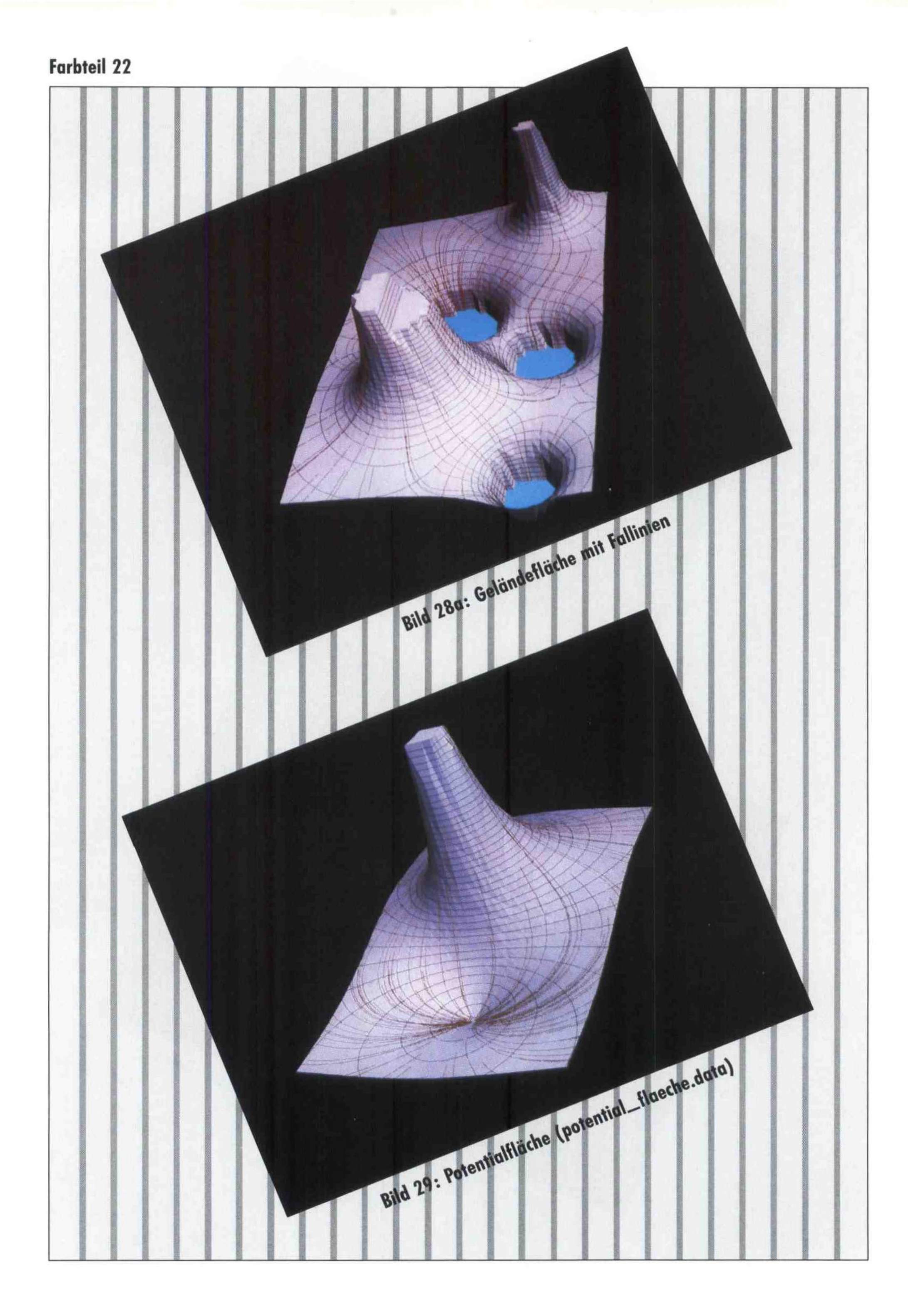


Bild 28a: Geländefläche mit Fallinien

Bild 29: Potentialfläche (potential_flaeche.data)

Bild 30: Gedämpfte Welle (welle1.data)

Bild 31: Wüstenlandschaft (wueste.data)

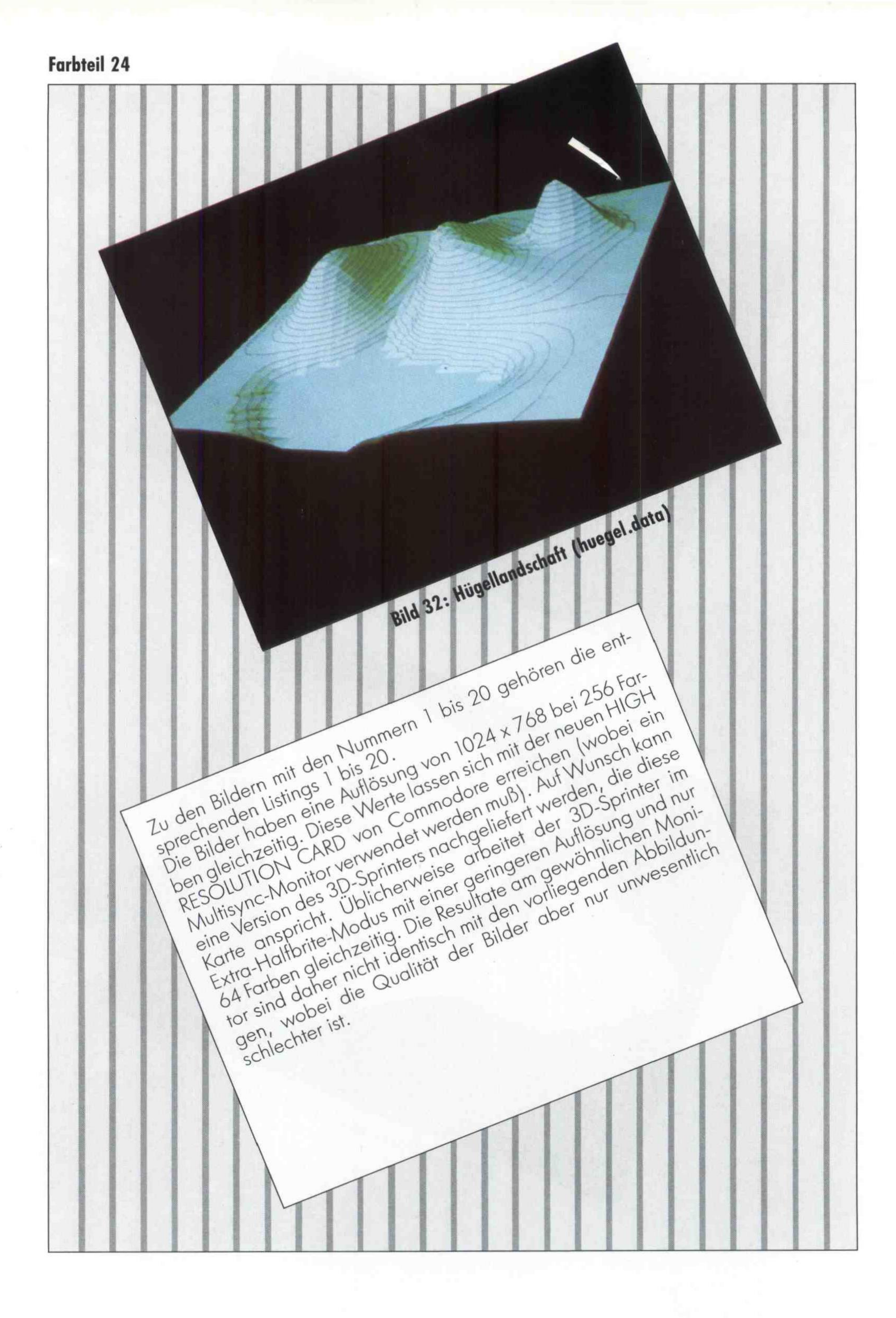


Bild 32: Hügellandschaft (huegel.data)

Zu den Bildern mit den Nummern 1 bis 20 gehören die entsprechenden Listings 1 bis 20. Die Bilder haben eine Auflösung von 1024 x 768 bei 256 Farben gleichzeitig. Diese Werte lassen sich mit der neuen HIGH RESOLUTION CARD von Commodore erreichen (wobei ein Multisync-Monitor verwendet werden muß). Auf Wunsch kann eine Version des 3D-Sprinters nachgeliefert werden, die diese Karte anspricht. Üblicherweise arbeitet der 3D-Sprinter im Extra-Halfbrite-Modus mit einer geringeren Auflösung und nur 64 Farben gleichzeitig. Die Resultate am gewöhnlichen Monitor sind daher nicht identisch mit den vorliegenden Abbildungen, wobei die Qualität der Bilder aber nur unwesentlich schlechter ist.

CIP-Titelaufnahme der Deutschen Bibliothek

Amiga 3D-Sprinter : interaktive Echtzeit-Animation ; Erzeugung und Bewegung von brillanten, spiegelnden Drahtgitter-, Plotter-, Schattierungs- und Schlagschattengrafiken. –
Haar bei München : Markt-u.-Technik-Verl., 1990
(Bookware)
ISBN 3-89090-109-3

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

93 92 91 90

ISBN 3-89090-109-3

Einbandgestaltung: Grafikdesign Heinz Rauner
Dieses Produkt wurde mit Desktop-Publishing-Programmen erstellt
und auf der Linotronic 300 belichtet
Druck: Schoder, Gersthofen
Printed in Germany

Inhaltsverzeichnis

Kapitel 5 Für Fortgeschrittene: Wir schreiben Animations-Skripten

Kapitel 6 Das Abspeichern und Abspielen von Animationen-»Videos«

Kapitel 7 Mehr Informationen über das Erstellen von Datenfiles

Kapitel 8 Ausplotten bzw. Ausdrucken der Zeichnungen

Kapitel 9 Wie entsteht so ein Programm eigentlich?

Anhang



Vorwort

Die 3-D-Grafik ist eines der komplexesten Anwendungsgebiete der Computertechnik. Es werden immer bessere und schnellere Verfahren zur Erstellung von immer realistischeren Grafiken entwickelt. Auf modernen Großrechenanlagen werden schon ganze Filme nur durch Berechnung hergestellt. Natürlich werden Computer in der Preisklasse des Amiga niemals an diese Leistungen herankommen, dennoch lassen sich erstaunliche Ergebnisse erzielen – so, wie auf dem Titelbild dieser Bookware dargestellt!

Überblick über das Softwarepaket und das Handbuch

Durch die Vielzahl von Objekttypen, die Ihnen der 3-D-Sprinter zur Verfügung stellt, können Sie mit etwas Übung sehr anschauliche Grafiken erstellen. Diese können Sie anschließend über die Tastatur, mit der Maus oder mit Hilfe von Animations-Skripten manipulieren (drehen, vergrößern, verkleinern, bewegen, . . .). Das Handbuch besteht aus zwei Abschnitten, die in neun Kapitel aufgeteilt wurden. Einer der beiden Abschnitte behandelt die Bedienung des Programms und den Umgang mit dem Amiga. Der andere beschäftigt sich mit der 3-D-Grafik. Um ein schnelles Arbeiten mit dem Programm zu ermöglichen, wurden die grundlegendsten Bedienungsschritte vorab in einem Kapitel zusammengefaßt.

Am Schluß beschreiben wir noch kurz, wie dieses Programm entstanden ist

Was unterscheidet den 3-D-Sprinter von anderen 3-D-Systemen? In den letzten Jahren entwickelte sich die 3-D-Grafik am Computer sehr rasant. Ein Meilenstein in dieser Entwicklung war das sogenannte »Raytracing«-Verfahren, das zuerst an Großcomputern getestet und mittlerweile voll auf grafiktaugliche Mikrocomputer wie den Amiga übertragen wurde. Die Qualität der Bilder ist bestechend, weil insbesondere Schlagschatten und Spiegelungen berechnet werden können. Demgegenüber steht der große und nur mit enorm teuren Transputer-Anlagen behebbare Nachteil: Die Berechnung der Bilder dauert zu lange, um wirklich effektiv arbeiten zu können.

Der 3-D-Sprinter verwendet verschiedene schnelle Verfahren zur Erstellung realistischer Bilder, die den Raytracing-Programmen an Rechengeschwindigkeit um Zehnerpotenzen überlegen sind. Der ursprünglich vermutete Nachteil, die Qualität der Bilder werde auch nicht annähernd an den Raytracing-Standard herankommen, läßt sich schon jetzt abschwächen: Der 3-D-Sprinter kann Schlagschatten einzeichnen und – wenn auch derzeit noch in den Kinderschuhen – Spiegelungen durchführen. Der Schritt zur Kombination Spiegelung-Schlagschatten ist nur

mehr sehr klein und wird demnächst durchgeführt werden. Noch ein weiterer Vorteil der im 3-D-Sprinter verwendeten Algorithmen: sie sind von der Auflösung des Bildschirms unabhängig!

Nur schnelle Programme erlauben das wirklich dreidimensionale Erfassen von Objekten am Bildschirm und in weiterer Folge die effektive Herstellung von längeren Trickfilmen. Dazu ist der 3-D-Sprinter bestens vorbereitet.

Wo so viele Vorteile sind, können aber auch Nachteile nicht fehlen: Nun, derzeit kann man die gewünschten Objekte nur mittels der eigens dafür entwickelten Sprache »3-D« eingeben. Dies ist für Leute, die von den nicht so benutzerfreundlichen Großcomputern kommen, kein besonderes Handicap. Der Amiga böte allerdings die Möglichkeit, Szenen mit sogenannten »Objekteditoren« zu erzeugen.

Weiterhin dürfen sich die Objekte nicht überschneiden, was manchmal recht lästig sein kann. Die nächste Version wird sicherlich einen »splitter« aufweisen, der nichtkonvexe Objekte in konvexe Teilbausteine »zerhackt«. Jedenfalls ist die Zeit, wo man sagen mußte, der 3-D-Sprinter sei nur für »einfache Szenen« geeignet, vorbei. Bei genügend Speicherplatz ist nach oben hin keine Grenze gesetzt. Dementsprechend sollte der 3-D-Sprinter Werbegrafiker, Architekten, Wissenschaftler und natürlich alle Grafikfreaks gleichermaßen ansprechen können.

Danksagungen

Natürlich kann ein so umfangreiches Softwarepaket, wie es der 3-D-Sprinter darstellt, nicht allein auf dem Mist von uns beiden gewachsen sein. Mit diesen Zeilen wollen wir uns bei all den vielen Menschen bedanken, die zur Realisation des Projektes beigetragen haben. Unser besonderer Dank gilt

- Prof. Silvio Levy vom Mathematischen Institut der Universität Princeton (USA) für seine zahlreichen Hinweise und seine aktive Hilfe beim Optimieren der Rechengeschwindigkeit;

- der Abteilung für Computergrafik an der Österreichischen Akademie der Wissenschaften (insbesondere Herrn Dipl.-Ing. Thaller) für die Bereitstellung eines hochauflösenden Grafikbildschirms (1024 x 768 Bildpunkte), um die Qualität der Abbildungen in diesem Buch zu verbessern;

- Commodore Österreich für die zahlreichen Unterstützungen;

- den österreichischen Amiga-Developern (insbesonders Max Hantsch) für ihre Hilfe beim Debuggen und Testen des Programms;

- Erich Grohser (Vater von Thomas) für die Hilfe bei der Korrektur des Handbuches;

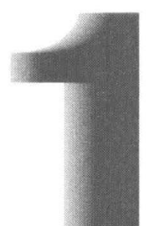
- den vielen Schülerinnen und Schülern der HTBL Wien X, die uns immer wieder Anregungen für neue 3-D-Grafiken gegeben haben (vor allem während einer Projektwoche, in der einfache Trickfilme mit den Datenfiles »Hubschrauber«, »Gebaeudekomplex« und »Fussgaengerbruecke« erstellt wurden). Dabei gab uns übrigens Wolfgang Schmid den Anstoß zum »Replay-Programm«, das die fertig berechneten Bilder abspeichert und danach schnell wieder abspielt;

- Jay Minner, Robert J. Mical, Dave Morse und ihren Mitarbeiterinnen und Mitarbeitern, dafür, daß sie den Traumcomputer Amiga gebaut und zum Leben erweckt haben;

- Andy Finkel, Bob »Kodiak« Burns, Jim Mackraz, Bryce Nesbitt, Carolyn Scheppner und ihren Mitarbeitern für ihre zahlreichen Hinweise auf der 2nd European Amiga Developers Conference, die dafür gesorgt haben, daß unser Programm auch mit zukünftigen Versionen des Amiga-Betriebssystems »Kickstart« laufen wird.

- Ganz besonders wollen wir uns auch noch bei Ihnen bedanken, daß Sie unseren Programmierkünsten vertraut und dieses Softwarepaket gekauft haben. Wir wünschen Ihnen beim Austesten der auf den Disketten vorhandenen Datenfiles sowie beim späteren kreativen Arbeiten viel Spaß.

Georg Glaeser & Thomas Grohser



Amiga-Grundbegriffe

Dieses Kapitel soll es den »Amiga-Neulingen« erleichtern, sich mit der Maschine so weit vertraut zu machen, daß sie unser Programm benutzen können. Wenn Sie sich mit dem Amiga, der Workbench, Menüs usw. bereits auskennen, können Sie, nachdem Sie die Sicherheitskopien der Programmdisketten erstellt haben, dieses Kapitel überspringen.

1.1 Einschalten und betriebsbereit machen

Wie Sie Ihren Amiga auspacken und aufstellen, entnehmen Sie bitte den beiliegenden Handbüchern. Dies zu erklären, würde bei der Vielzahl von Amiga-Modellen den Rahmen dieses Kapitels bei weitem sprengen.

Nachdem Sie den Amiga 500 oder 2000 eingeschaltet haben, meldet er sich mit der Aufforderung, die Workbench-Diskette einzulegen. Amiga-1000-Besitzer müssen zuerst noch das Betriebssystem Kickstart laden, da es bei diesem Rechnermodell noch nicht fest eingebaut ist. Legen Sie die Kickstart-Diskette in das interne Disketten-Laufwerk. Verwenden Sie dazu bitte nur Kickstart-Disketten mit der Version 1.2 oder höher.

Nun melden sich auch die »alten Amigas« mit der Aufforderung, die Workbench einzulegen. Kommen Sie dieser Aufforderung nach und legen Sie die Programmdiskette in das Diskettenlaufwerk mit der Bezeichnung »DF0:«. Beim Amiga 500 und 1000 ist dies das interne Disketten-Laufwerk, beim Amiga 2000 ist es normalerweise das Laufwerk ganz rechts. Nach einigen Sekunden sieht der Bildschirm so aus:

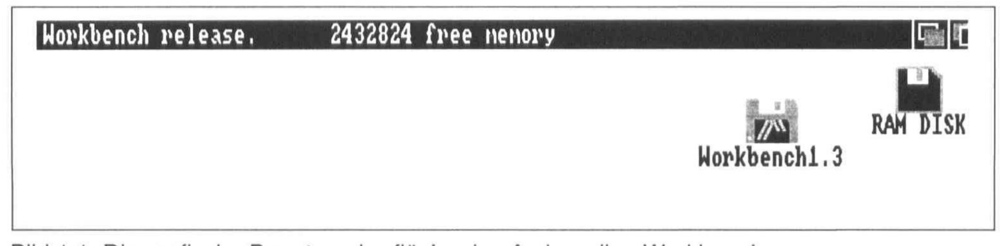


Bild 1.1: Die grafische Benutzeroberfläche des Amiga, die »Workbench«

Sie befinden sich jetzt auf der Workbench, der grafischen Benutzeroberfläche des Amiga. Die Anzeige des freien Speichers (in der obersten Bildschirmzeile) ist je nach Ausbaustufe natürlich anders.

1.2 Sicherheitskopien der Programmdisketten

Bevor wir mit dem Arbeiten beginnen, sollten Sie prinzipiell Sicherheitskopien der Programmdisketten machen. Halten Sie bitte zwei leere Disketten bereit. Die Frage, welche Disketten die besten sind, ist schwer zu beantworten. Wir empfehlen Ihnen jedoch, nicht am falschen Platz zu sparen. Kaufen Sie auf jeden Fall nur doppelseitige Disketten mit doppelter Schreibdichte (Disketten, die diese Eigenschaften besitzen, werden mit den Abkürzungen DSDD oder 2S2D beschriftet). Wenn Sie qualitativ schlechtere Disketten benutzen, kann es leichter passieren, daß die Diskette schadhaft wird und dadurch die Daten, die Sie auf ihr speichern, verlorengehen.

Natürlich gibt es auch keine Garantie dafür, daß eine Qualitätsdiskette nicht defekt sein kann oder es mit der Zeit wird (keine Diskette der Welt hält ewig!). Aus diesem Grund sollten Sie in regelmäßigen Abständen Sicherheitskopien Ihrer Disketten anfertigen. Je nachdem, ob Sie ein oder zwei Disketten-Laufwerke besitzen, läuft der Vorgang anders ab.

1.2.1 Sicherheitskopien mit einem Diskettenlaufwerk

Zuerst kopieren wir die Programmdiskette, die zugleich eine Amiga-Workbench-Diskette ist. Bewegen Sie den Maus-Pfeil auf das Piktogramm, unter dem der Text »WorkBench1.3« steht, und betätigen Sie einmal die linke Maus-Taste. Nun sollte sich das Aussehen des Piktogramms verändert haben:

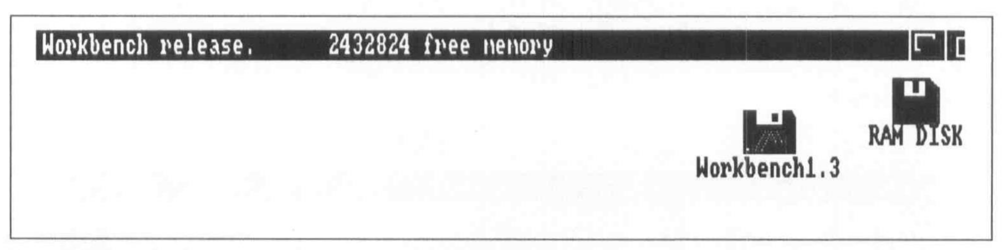


Bild 1.2: Anwählen eines Piktogramms (Icon)

Bewegen Sie jetzt den Maus-Pfeil in die linke obere Ecke des Bildschirms. Drücken Sie die rechte Maus-Taste und halten Sie sie gedrückt. Es erscheint ein Menü, das folgendermaßen aussieht:

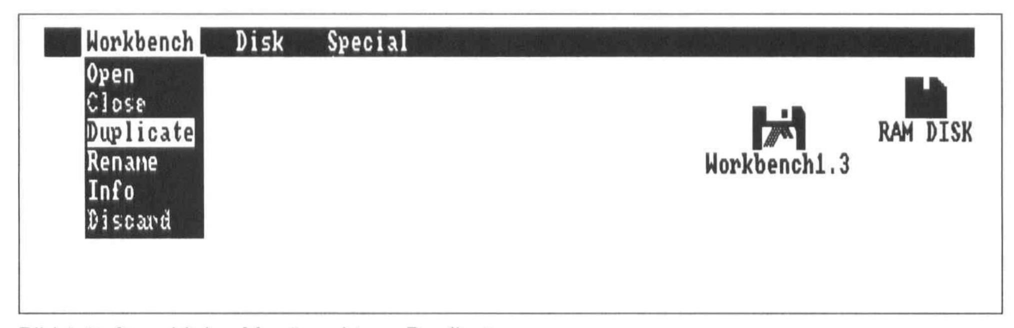


Bild 1.3: Anwahl des Menüpunktes »Duplicate«

Bewegen Sie nun den Maus-Pfeil in die Zeile, in der »Duplicate« steht, und lassen Sie die rechte Maus-Taste wieder los. Nach einigen Sekunden Ladezeit erscheint das folgende Kommunikationsfenster (Requester). Dieses fordert Sie auf, die Quelldiskette »FROM-Disk« einzulegen.

Da sich diese bereits im Laufwerk befindet, bewegen Sie den Maus-Pfeil über das »Continue«-Symbol und betätigen die linke Maus-Taste. Nun liest der Amiga einen Teil der Diskette in den Speicher. Danach fordert er Sie auf, die Ziel-Diskette einzulegen. Das Kommunikationsfenster sieht aus wie das vorherige, nur daß aus dem Text »FROM-Disk« ein »TO-Disk« geworden ist. Kommen Sie dieser Aufforderung nach. Jetzt schreibt der Amiga die Daten auf die neue Diskette. Im Anschluß daran müssen Sie die Quelldiskette wieder einlegen. Dieser Vorgang wiederholt sich je nach Größe des freien Speichers mehrmals.

Nachdem die Diskette kopiert ist, erscheint auf dem Bildschirm ein weiteres Piktogramm. Unter diesem steht der Text »copy of Workbench 1.3«. Aktivieren Sie dieses Piktogramm, indem Sie den Maus-Pfeil darüber bewegen und die linke Maus-Taste drücken. Diesen Vorgang nennt man auch »Anklicken«.

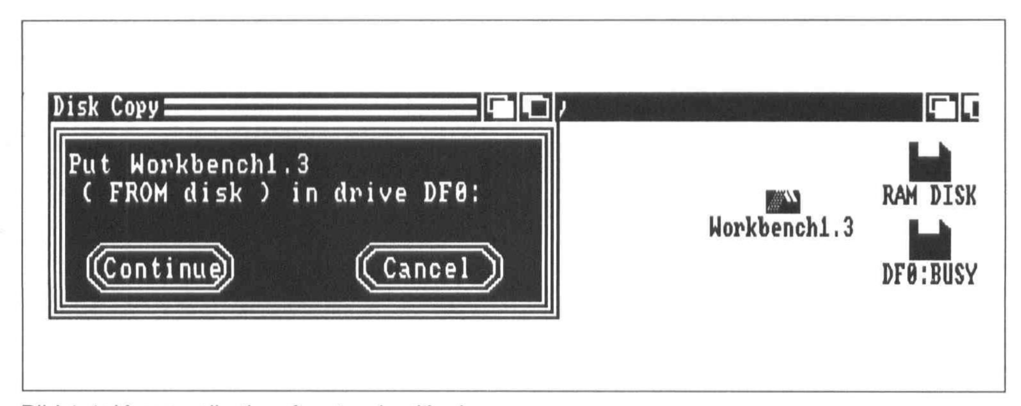


Bild 1.4: Kommunikationsfenster des Kopierprogramms

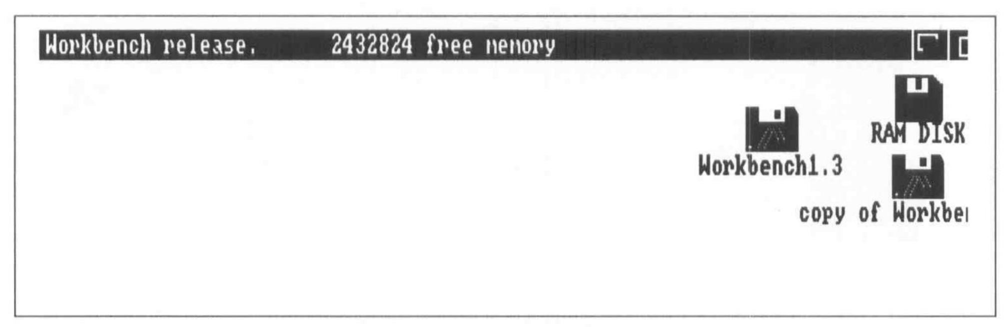


Bild 1.5: Anklicken eines Piktogramms

Wählen Sie jetzt im Menü der Workbench den Menüpunkt »Workbench->Rename« aus, indem Sie den Maus-Pfeil in die linke obere Ecke bewegen, die rechte Maus-Taste drücken, danach den Mauszeiger über die Zeile, in der Rename steht, bewegen und die rechte Maus-Taste wieder loslassen. Dieser Vorgang wird auch das »Anwählen eines Menüpunktes« genannt.

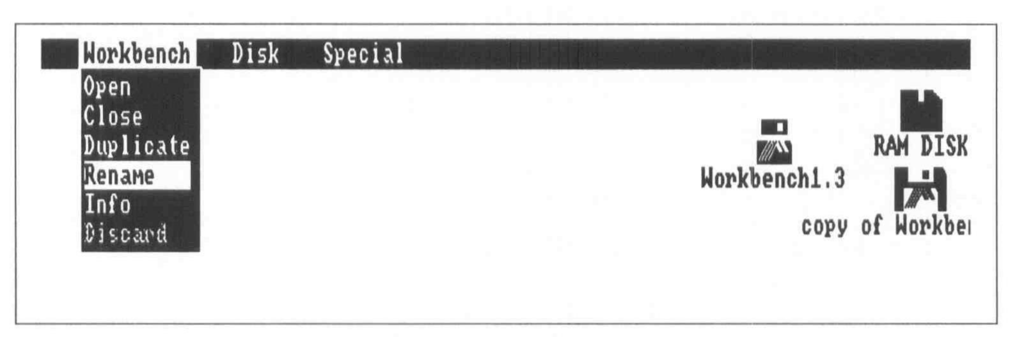


Bild 1.6: Anwählen eines Menüpunktes

Danach erscheint in der Mitte des Bildschirmes ein Rechteck, in dem der Text »copy of Workbench 1.3« steht. Bewegen Sie den Cursor mit den Cursor-Tasten an den Anfang des Textes und löschen Sie mit der [Del]-Taste alle Buchstaben und Leerzeichen, bis nur mehr der Text »Workbench 1.3« übrig bleibt. Betätigen Sie jetzt die [Return]-Taste, um den Vorgang des Umbenennens zu beenden.

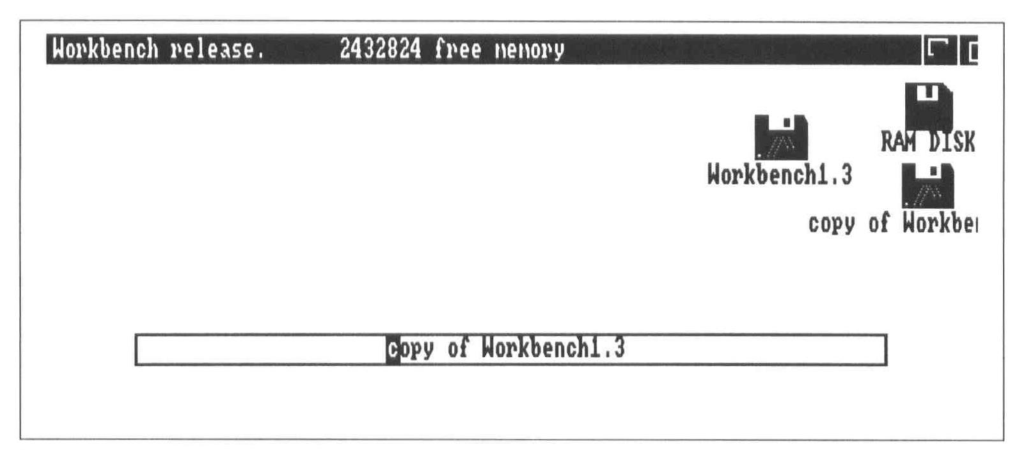


Bild 1.7: Das Umbenennen eines Piktogramms

Nehmen Sie jetzt die Diskette aus dem Laufwerk, und beschriften Sie sie mit dem Programmnamen (in diesem Fall mit »3-D-Sprinter-Programmdisk«). Ab jetzt sollten Sie mit der Kopie arbeiten und das Original an einem sicheren Ort verwahren. Um eine Diskette zu kopieren, die keine Workbench-Diskette ist, muß ein klein wenig anders vorgegangen werden:

Dies resultiert aus der Tatsache, daß der Amiga das Kopierprogramm zuerst von der Workbench-Diskette laden muß. Legen Sie die Diskette, die Sie kopieren wollen (in unserem Fall ist das die Datendiskette), in das interne Laufwerk, warten Sie, bis das Piktogramm der Diskette erscheint, und aktivieren Sie es, indem Sie es anklicken (linke Maus-Taste).

Wählen Sie die Funktion »Duplicate« im Workbench-Menü an (rechte Maus-Taste festhalten, Menü aktivieren und Maus-Taste loslassen). Nun verlangt der Amiga im Gegensatz zu vorher, daß Sie die Workbench-Diskette einlegen. Kommen Sie dieser Aufforderung nach. Nach einigen Sekunden müssen Sie dann wieder die Diskette, die Sie kopieren wollen, in das interne Laufwerk legen. Ab jetzt verläuft alles wie oben beschrieben.

1.2.2 Sicherheitskopien mit zwei Diskettenlaufwerken

Zuerst werden wir die Programmdiskette, die zugleich eine Amiga-Workbench-Diskette ist, kopieren. Legen Sie in das erste (interne) Laufwerk (»Df0:«) die Original-Diskette (wenn Sie bis jetzt der Anleitung gefolgt sind, müßte sich die Programmdiskette eigentlich schon im Laufwerk befinden) und in das zweite (externe) Laufwerk die leere Diskette. Nun erscheinen auf der Workbench zwei Piktogramme. Bewegen Sie den Maus-Pfeil über das Piktogramm der Originaldiskette (unter dem Piktogramm steht »Workbench 1.3«), und drücken Sie die linke Maus-Taste. Das Piktogramm ändert jetzt sein Aussehen; halten Sie die Maus-Taste gedrückt,

bewegen Sie das Piktogramm über das der leeren Diskette, und lassen Sie die Maus-Taste wieder los. Nach einiger Zeit erscheint das folgende Kommunikationsfenster:

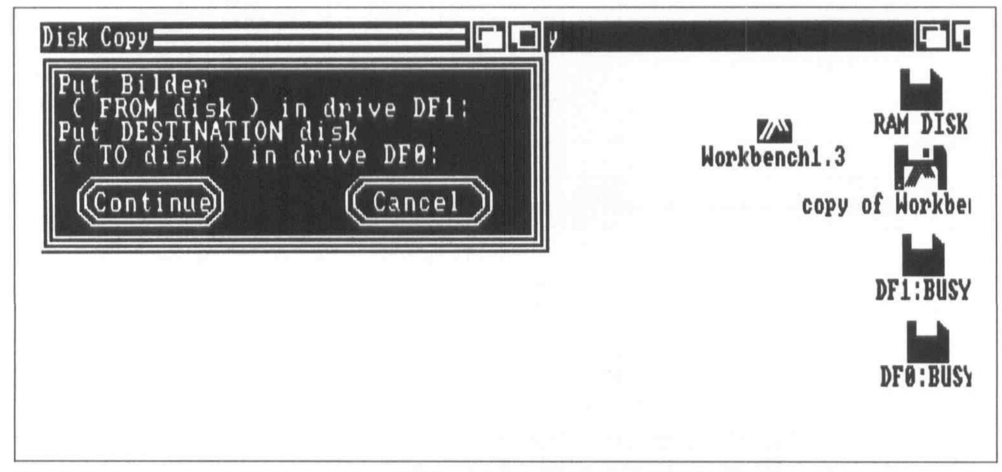


Bild 1.8: Kommunikationsfenster zum Diskettenkopieren mit zwei Laufwerken

Beantworten Sie die Aufforderung des Betriebssystems, indem Sie den Maus-Pfeil über das »Continue«-Symbol bewegen und dort die linke Maus-Taste betätigen. Nun beginnt der Kopiervorgang. Wenn er abgeschlossen ist, muß der Name der Kopie noch geändert werden. Wie dies vor sich geht, lesen Sie auf Seite 16.

Soll eine Diskette kopiert werden, die keine Workbench-Diskette ist, wird man, bevor das oben gezeigte Kommunikationsfenster erscheint, aufgefordert, die Workbench-Diskette in ein Laufwerk einzulegen. Nachdem der Amiga das Kopierprogramm geladen hat, muß dann wieder die zu kopierende Diskette eingelegt werden. Der Rest des Kopiervorgangs verläuft wie oben beschrieben.

Nach dem Kopieren der beiden Disketten sollten Sie den Amiga durch Drücken der drei Tasten [Ctrl]+[RAmiga]+[LAmiga] in den Einschaltzustand zurückversetzen und ihn mit der Kopie der Programmdiskette neu starten. Die Originale sollten Sie an einem sicheren Ort aufbewahren.

1.3 Von Menüs, Piktogrammen und Symbolen

Beim Herstellen der Sicherheitskopie der Programm- und Datendiskette (die Sie hoffentlich bereits angefertigt haben, wenn nicht, dann aber schnell) haben Sie schon die grundlegende Bedienungstechnik des Amiga kennengelernt.

a) Aktivieren von Piktogrammen und Symbolen (linke Maus-Taste)
b) Anwählen von Menü-Punkten (rechte Maus-Taste)

Jede Diskette wird beim Amiga durch ein Symbol (Piktogramm) dargestellt. Um sich den Inhalt einer Diskette anzusehen, muß man den Mauszeiger auf das Symbol bewegen und zweimal schnell hintereinander die linke Maus-Taste betätigen (dieser Vorgang wird auch »Doppelklick« genannt).

Danach öffnet sich ein Fenster, dessen Inhalt zum Beispiel so aussehen kann (so sieht der Inhalt der Programmdiskette aus):

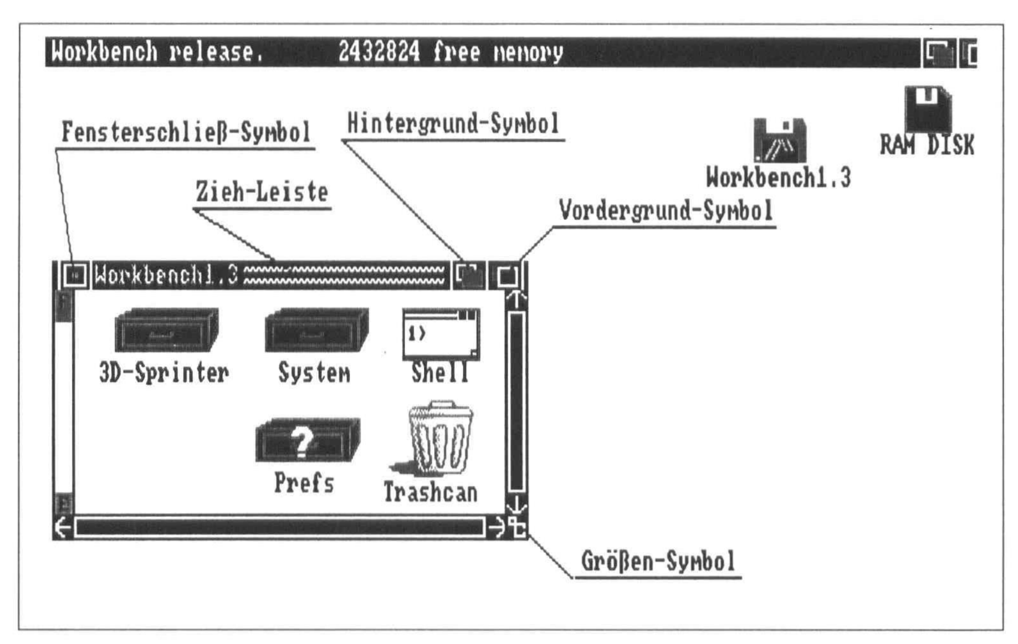


Bild 1.9: Der Inhalt der Programmdiskette des 3-D-Sprinters

1.4 Fenster

Wenn wir das Fenster näher betrachten, so finden wir darin und auf dem Rahmen eine Vielzahl von Symbolen. Auf der vorherigen Abbildung sind die Namen der einzelnen Symbole eingezeichnet, die nachfolgend erklärt werden:

Fensterschließ-Symbol: Mit diesem Symbol kann man, wie der Name schon sagt, das Fenster schließen (ein einfacher Mausklick genügt).

Zieh-Leiste: Damit kann das Fenster verschoben werden. Klicken Sie in den Bereich der Leiste, und halten Sie die linke Maus-Taste gedrückt. Bewegen Sie

jetzt das Fenster an die Stelle, an der es nachher erscheinen soll, und lassen Sie die Maus-Taste wieder los.

Vordergrund-Symbol: Wenn mehrere Fenster geöffnet sind, wird das Fenster nach einem Klick auf dieses Symbol als oberstes dargestellt.

Hintergrund-Symbol: Bewirkt genau das Gegenteil, das Fenster wird als hinterstes dargestellt.

Größen-Symbol: Damit wird die Größe des Fensters verändert. Solange die linke Maus-Taste über dem Symbol gedrückt gehalten wird, kann die Größe verändert werden.

Im Inneren des Fensters sind wieder Symbole zu sehen, einige davon sehen aus wie Schubladen. Den Inhalt einer Schublade kann man sich genauso wie den einer Diskette ansehen.

1.5 Screens – Bildschirme

Wir haben eigentlich schon die ganze Zeit auf einem Screen (dem »Workbench Screen«) gearbeitet. Wieso das beim Amiga etwas Besonderes ist? Der Amiga kann auch mehrere Screens darstellen. Die Position eines Screens kann man verändern, wenn man den Maus-Pfeil in die Titel-Leiste bewegt, die linke Maus-Taste betätigt und gedrückt hält und dann die Maus bewegt. Welcher Bildschirm als vorderster erscheinen soll, kann mit den Vordergrund- und Hintergrund-Symbolen, die genauso wie bei den Fenstern funktionieren, festgelegt werden.

1.6 Programme starten

Um ein Programm zu starten, muß das dem Programm zugehörige Symbol schnell zweimal hintereinander mit der linken Maus-Taste angeklickt werden. Um das auszuprobieren, öffnen Sie die Schublade »Prefs« und starten das Programm »Preferences«.

1.7 Voreinsteller (das Preferences-Programm)

Das Preferences-Programm ist ein sehr wichtiges und nützliches. Mit ihm können Sie den verwendeten Drucker, die Bildschirmfarben, die Maus-Geschwindigkeit u.v.m. einstellen. Leider ändert sich der Aufbau des Preferences-Programms von einer Betriebssystemversion zur anderen.

Deshalb hat eine genaue Erklärung nicht viel Sinn, ziehen Sie dazu die dem Amiga beiliegende Beschreibung hinzu. Es sei nur erwähnt, daß die grafische Benutzer-

oberfläche des 3-D-Sprinters die in Preferences eingestellten Farben benutzt und die Einstellungen des Druckers berücksichtigt.

Beendet wird das Preferences-Programm durch einen Klick in das »Cancel«-Symbol – wenn Sie die Einstellungen, die vor dem Aufruf des Programms aktiv waren, beibehalten wollen – oder durch einen Klick in das »Save«-Symbol, wenn Sie die neuen Einstellungen übernehmen wollen. Natürlich konnte diese kurze Einführung in die Bedienung des Amiga nicht das Studium der Handbücher bzw. der entsprechenden Literatur (siehe Anhang F) ersetzen, aber für den Anfang müßte es genügen. Wenn Sie sich eingehend mit dem Amiga beschäftigen wollen, sollten Sie auf jeden Fall vorher die Handbücher genau lesen.

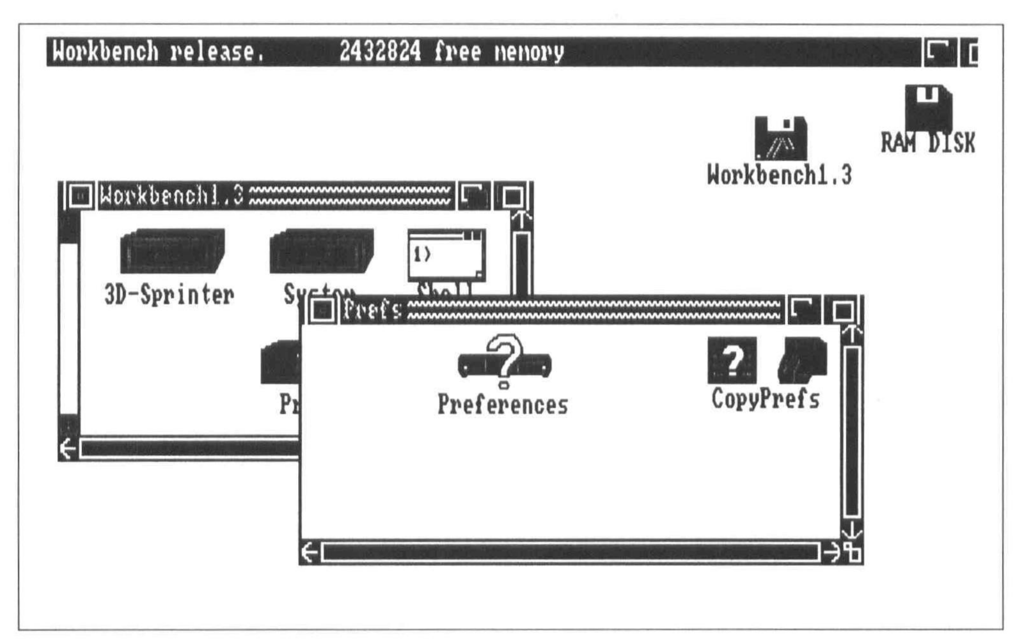


Bild 1.10: Die Piktogramme des Preferences-Programms

1.8 Sie haben eine 2620-Beschleunigerkarte?

Wenn Sie tatsächlich eine Turbokarte vom Typ 2620 besitzen, dann verwandelt sich der 3-D-Sprinter in einen 3-D-»Hurricane«. Das schnelle RAM des 68020-Prozessors macht den Sprinter »aus dem Stand« 2- bis 3mal schneller, aber erst der 68881-Mathematik-Coprozessor bringt die volle Beschleunigung zum Tragen. Im Durchschnitt wird der 3-D-Sprinter insgesamt um das Sechsfache schneller, besonders »mathematische« Routinen, die vor allem im Hidden-Line-Modus angewandt werden, werden sogar noch rascher abgearbeitet. Da hat selbst die super-

schnelle Grafik-Hardware des Amiga ihre Not, mitzukommen! Die »Dorfkirche« (Farbbild 10) oder die »Würfel vor Spiegel« (Farbbild 20) sind jetzt in kaum einer Sekunde mit Schlagschatten bzw. Spiegelung dargestellt, selbst die fünf »Pokerwürfel« mit immerhin 3000 Punkten (Farbbild 15) werden innerhalb von 3 Sekunden auf dem Spiegeltablett serviert.

Um den Coprozessor auch wirklich ansprechen zu können, müssen Sie nur das Programm »try«, das sich (aus Platzmangel) auf der mitgelieferten Datendiskette im Verzeichnis 2620 befindet, in das Verzeichnis 3 D-SPRINTER Ihrer Programm-Diskette kopieren. Vorsicht: Dabei überschreiben Sie die Normalversion von »try«! Aber Sie haben ja ohnehin Sicherheitskopien Ihrer Disketten angelegt. Wenn Sie ein bißchen erfahrener sind (und das sind Sie, wenn Sie eine Beschleunigerkarte Ihr eigen nennen), eröffnen Sie eine Shell (ein CLI-Fenster) durch Anklicken des entsprechenden Piktogramms und schreiben Sie die Kommandozeile

```
copy  Datendisk_1:2620/try  Workbench1.3:3 D-Sprinter
```

in das Fenster.

Vorsicht: Der 3-D-Sprinter läuft jetzt nur mehr mit Beschleunigerkarte. Wenn Sie keine solche Karte haben, versuchen Sie das ganze lieber nicht. Ein Guru-Gruß wäre die zwingende Folge davon!



Zum Aufwärmen wird eine Kaffeekanne dargestellt

Wenn man stolzer Besitzer eines neuen Programms ist, möchte man dieses natürlich gleich einmal ausprobieren. Deshalb wird im folgenden Kapitel kurz beschrieben, wie eine bereits erstellte Animation geladen und dargestellt werden kann. Außerdem ist diese Einführung in die Bedienung des Programms sicher beim Lesen von Kapitel 3 nützlich, da Sie dann die angegebenen Beispiele gleich ausprobieren können.

2.1 Starten des Programms

Um das Programm zu starten, muß die Schublade, in der sich das Programm befindet, geöffnet werden (unter ihrem Piktogramm steht 3-D-Sprinter). In dieser Schublade befinden sich einige Piktogramme, die für Programme stehen. Die grafische Benutzeroberfläche wird durch zweimaliges Anklicken des Piktogramms, unter dem 3-D-Sprinter steht, gestartet (vgl. Bild 2.1).

Nach einer kurzen Ladezeit wird ein neuer Bildschirm eröffnet und einige Zeilen mit Copyright-Meldungen ausgegeben. Um den Anwender nicht zu langweilen, kann dieser Vorgang jederzeit abgebrochen werden: Sobald die Maus in den Bereich der Titel-Leiste des Bildschirms bewegt wird, wird der Bildschirm gelöscht, und man kann mit der Arbeit beginnen.

2.2 Laden des Datenfiles

Das Programm wird ausschließlich über Menüs und Kommunikationsfenster bedient (für die Tastatur-Freaks existieren zusätzlich für fast alle Menüpunkte auch Tastaturabkürzungen).

Wir haben uns bei der Erstellung der Menüs an die von Commodore ausgegebenen Richtlinien gehalten, deshalb dürfte es kaum Schwierigkeiten geben, sich in den Menüs zurechtzufinden, wenn man bereits mit anderen Programmen auf dem Amiga gearbeitet hat. Alle Lade- und Speicheroperationen werden mit dem Datei-Menü durchgeführt (vgl. Bild 2.2).

Bild 2.1: Die Schublade mit den Programmen des 3-D-Sprinters

Wählen Sie zuerst den Menüpunkt Datei->Laden->Datenfile an.

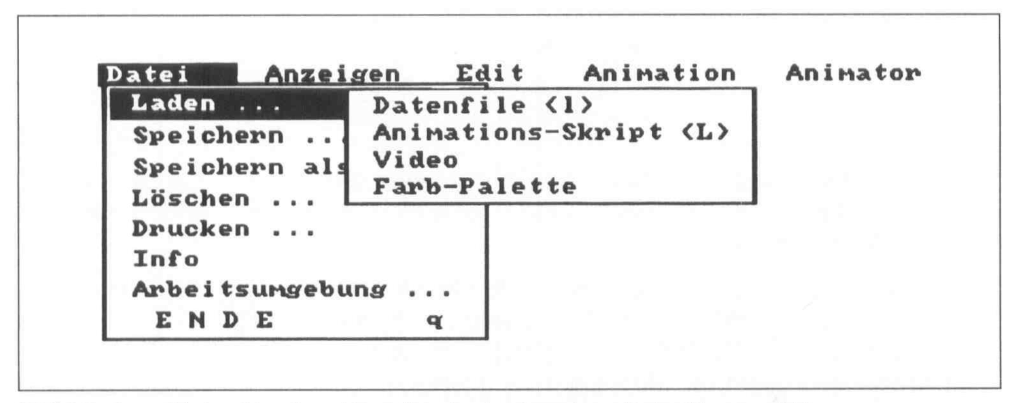


Bild 2.2: Anwahl des Menüpunktes, der das Laden eines Datenfiles bewirkt

Auf dem Bildschirm sollte sich folgendes Kommunikationsfenster geöffnet haben.

Im oberen Drittel des Kommunikationsfensters werden die ersten acht Datenfiles, die als Beispiele mitgeliefert werden, angezeigt. Weiter können durch Verändern der Position des Schiebereglers neben den Dateinamen weitere sichtbar gemacht werden (wem dies zu schnell ist, der kann mit den beiden Pfeilen über und unter dem Schieberegler die Liste der Dateien Zeile für Zeile verschieben).

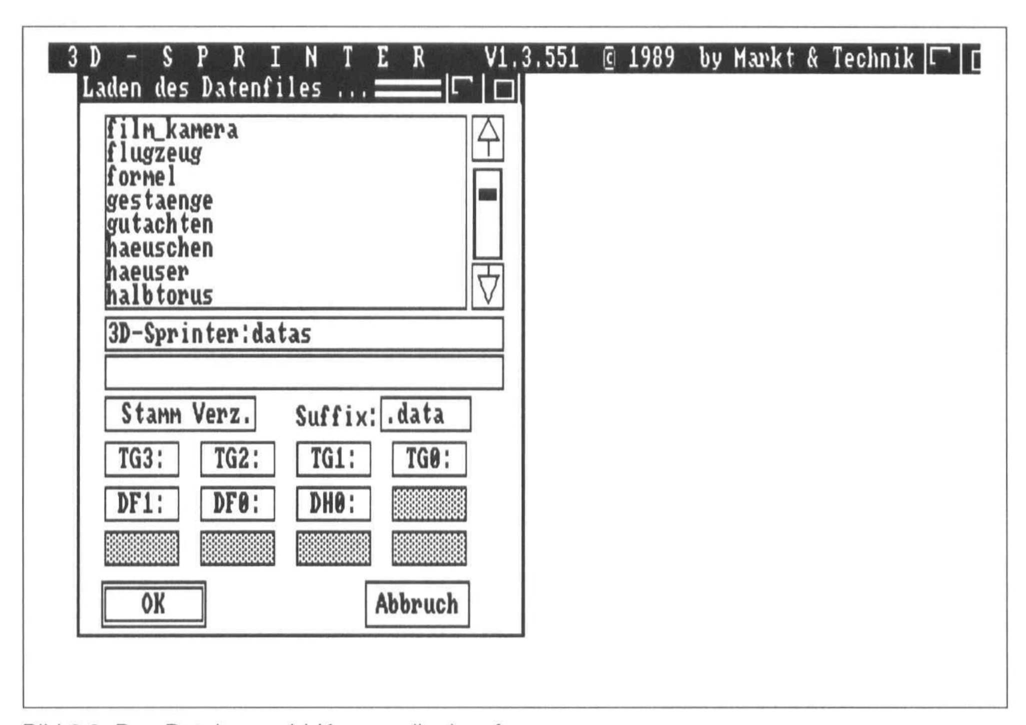


Bild 2.3: Das Dateiauswahl-Kommunikationsfenster

Verschieben Sie den Regler so lange, bis Sie eine Datei mit dem Namen »Kaffee« entdecken. Bewegen Sie den Maus-Pfeil über diese Zeile und klicken Sie einmal mit der linken Maus-Taste. Der Dateiname wird jetzt unterhalb in einer separaten Zeile dargestellt. Sollten Sie sich in der Zeile geirrt haben, klicken Sie einfach in die richtige Zeile. Wenn der richtige Dateiname (Kaffee) ausgewählt wurde, klicken Sie in das »O.K.«-Symbol. Das Kommunikationsfenster schließt sich wieder, und der Maus-Zeiger ändert kurz sein Aussehen (aus dem rotierenden Schriftzug wird die Schrift »BUSY«).

2.3 Starten der Animation

Um das geladene Datenfile jetzt darzustellen, wählen Sie bitte den Menüpunkt »Animation->Start ...->Autorotation« an.

Es werden nun die Grafikroutinen nachgeladen, und die Berechnung der Grafik beginnt. Jetzt müssen vorab einige Worte über die Darstellungsweise der Grafik gesagt werden.

```
Datei     Anzeigen     Edit     Animation     Animator
                                Start ...   | Interaktiv             I
                                Zeichenmod  | Autorotation           A
                                  Distanz-  | Animations-Skript      S
                                veränderun  | Autorot -> Video       V
                                Rotation u  | Skript -> Video        W
                                            | Video Replay           R
```

Bild 2.4: Anwahl des Menüpunktes »Animation->Start->Autorotation«

Das Programm versucht, den Animationseffekt durch sogenanntes »Double Buffering« bzw. »Page-Flipping« zu erzeugen. Dazu muß allerdings genug Speicher (CHIP-RAM) vorhanden sein. Es können nun drei Fälle eintreten:

a) Sie haben genug Speicher – auf dem Bildschirm beginnt sich eine Kaffeekanne zu drehen.

b) Sie haben genug Speicher, um einen Bildschirm zu eröffnen. Es wird das erste Bild berechnet und dargestellt. Nach einem Druck auf die Leertaste das nächste, usw. ...

c) Sie haben nicht genug Speicher – kein Bildschirm öffnet sich; lesen Sie in diesem Fall bitte in Kapitel 4.6.1 nach, wie Sie die Bildschirmauflösung der Animatorbildschirme verringern können. Danach sollte alles klargehen (Fall »c«) dürfte eigentlich nur auf Amigas ohne Speichererweiterung passieren).

Die Körper werden aber alle noch als Drahtgittermodell dargestellt. Drücken Sie die Taste [h] (wie »hidden lines«) auf der Tastatur, und alle verdeckten Linien werden entfernt. Dieser Vorgang erhöht die Anschaulichkeit der Szene schon gewaltig. Wenn Sie jetzt die Taste [s] (wie »shades«) betätigen, werden die Flächen eingefärbt. Und um die Bildqualität noch weiter zu erhöhen, drücken Sie doch einmal die Taste [c] (wie »cast shadows«). Jetzt werden auch noch Schatten eingezeichnet! Mit der Taste [w] (wie »wire frame«) kommen Sie jederzeit wieder in den Drahtgittermodus zurück. Genaueres über die verschiedenen Darstellungsarten erfahren Sie in Kapitel 3.4. Wenn Sie genug mit den vier Darstellungstech-

niken, die das Programm besitzt, gespielt haben, kommen Sie durch Betätigen der [q]- oder [Esc]-Taste wieder zu den Menüs zurück.

2.4 Laden des Animations-Skripts

Bis jetzt konnten wir die Bewegung des Körpers nicht beeinflussen. Die Kaffeemaschine dreht sich immer nur im Kreis. Man kann zwar über einen Menüpunkt die Drehrichtung festlegen, richtig schön wird es aber erst, wenn man die Bewegung der Kamera (des Auges), des Lichtes u. v. m. in einem Animations-Skript festhält und dieses dann abarbeiten läßt. Um es Ihnen am Anfang nicht zu schwer zu machen, haben wir auch einige Animations-Skripten zum Ausprobieren auf die Diskette gespielt.

Eines der Animations-Skripten der Kaffeemaschine wollen wir jetzt laden und dann die Animation von ihm gesteuert ausführen lassen. Wählen Sie dazu bitte den Menüpunkt »Datei->Laden->Animations-Skript« an.

Es öffnet sich wieder das Kommunikationsfenster, das wir vom Laden des Datenfiles kennen. Wählen Sie das Animations-Skript mit der Bezeichnung »Kaffee_1« aus und beenden Sie die Auswahl mit einem Klick in das »O.K.«-Symbol. Starten Sie die Animation jetzt durch Anwahl des Menüpunktes »Animation->Start ... ->Animations-Skript«.

Jetzt sollte sich die Kaffeekanne in alle möglichen Richtungen drehen. Wenn sich nichts tut, haben Sie zuwenig Speicher (reduzieren Sie die Bildschirmauflösung; wie dies vor sich geht, können Sie in Kapitel 4.6.1 nachlesen). Auch in diesem Modus wird die Animation durch [q] bzw. [Esc] beendet. Ansonsten läuft sie bis zum Ende durch.

2.5 Starten und Speichern als Video

Um eine Szene als Video abzuspeichern, muß zuerst der Name, unter dem sie abgespeichert werden soll, angegeben werden. Wählen Sie hierfür den Menüpunkt »Datei->Speichern als ...->Video« an. Aktivieren Sie mit dem Maus-Pfeil und der linken Maus-Taste das Eingabefeld für den Dateinamen und geben Sie »Kaffee_1« ein. Wichtig ist, daß Sie keine Leerzeichen, sondern Unterstriche ([_] bzw. [Shift]+[-]) in den Dateinamen verwenden.

Bestätigen Sie die Eingabe durch Anklicken des »O.K.«-Symbols. Nun können Sie die Animation durch Anwählen des Menüpunktes »Animation->Start ...->Skript ->Video« starten. Jetzt werden alle Einzelschritte der Animation berechnet und abgespeichert. Ein vorzeitiger Abbruch ist wie immer mit [q] oder [Esc] möglich.

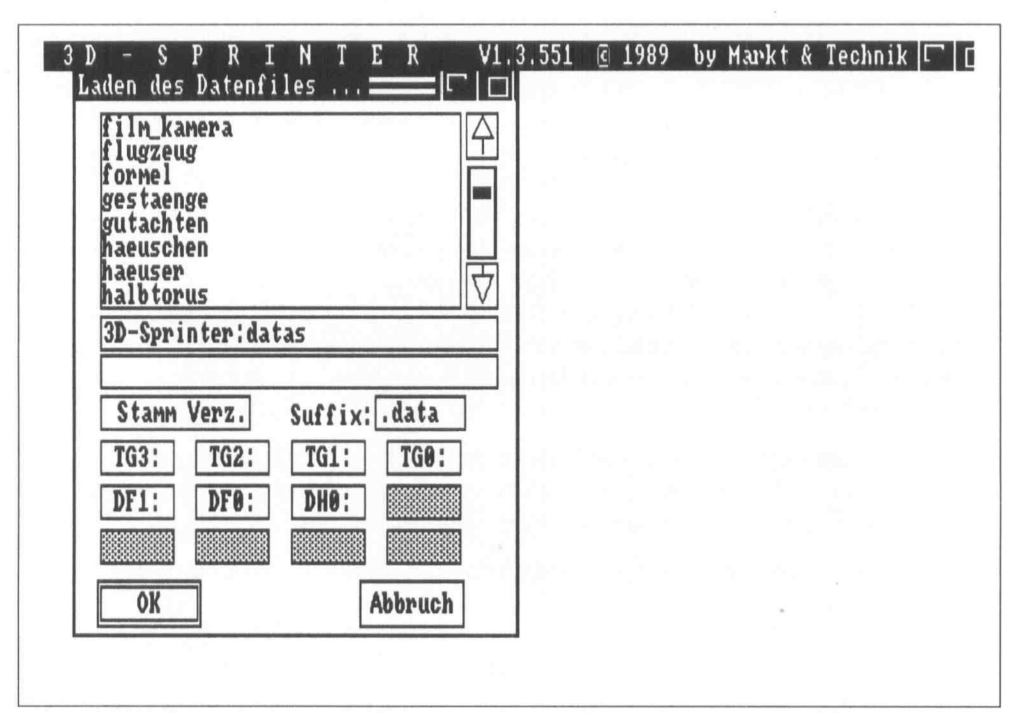


Bild 2.5: Das Eingabefeld für Dateinamen

2.6 Wiederabspielen des Videos

Um das Video wieder abzuspielen, wählen Sie den Menüpunkt »Animation->Start ...->Video Replay« an. Nun kann es je nach Länge der Animation einige Zeit dauern, da zuerst alle Daten in den Speicher geladen werden. Danach wird die Animation schnell abgespielt. Auch hier geschieht der Abbruch mit [q] oder [Esc].

2.7 Manipulieren des Datenfiles

Um ein Datenfile oder Animations-Skript zu erstellen oder zu verändern, benötigen Sie einen Text-Editor. Um nicht ständig zwischen dem 3-D-Sprinter und dem Text-Editor hin- und herwechseln zu müssen, ruft der 3-D-Sprinter den Text-Editor auf, wenn er ihn benötigt. Standardmäßig nimmt der 3-D-Sprinter den »Ed« von der Workbench. Wie Sie Ihren Lieblings-Editor einstellen können, erfahren Sie in Kapitel 4.2.8. Der Grund, warum wir dies bereits hier erwähnen, ist, daß Sie in der Lage sein sollen, im nächsten Kapitel alle Beispiele ausprobieren zu können. Wenn Sie ein Datenfile edieren wollen, wählen Sie den Menüpunkt »Edit->Datenfile« an. Für Animations-Skripten ist »Edit->Animations-Skripten« zuständig.

2.8 Beenden des Programms

Um das Programm zu beenden, müssen Sie den Menüpunkt »Datei ->Ende« anwählen. Es erscheint ein Kommunikationsfenster, in dem Sie darauf hingewiesen werden, daß beim Verlassen des Programms Daten, die Sie noch nicht abgespeichert haben, verlorengehen können.

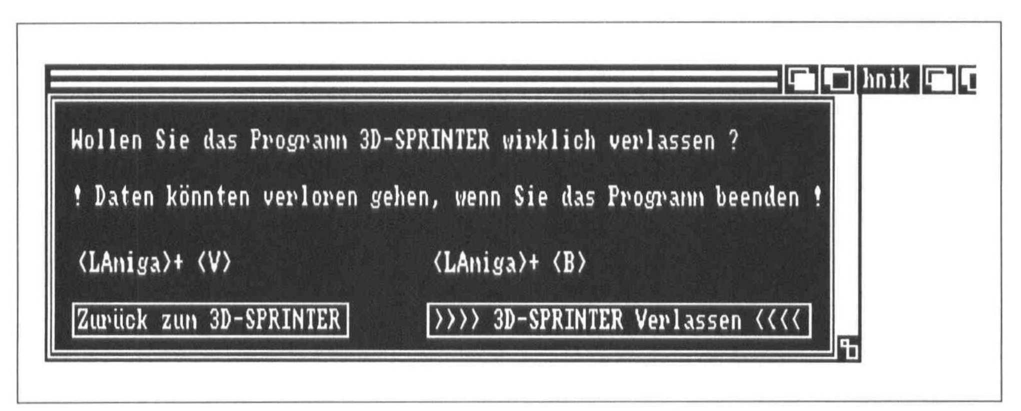


Bild 2.6: Sicherheitsabfrage vor dem Programmende

Durch Anwahl des »3-D-Sprinter verlassen«-Symbols wird das Programm dann endgültig beendet.

3 Was man über die Theorie der 3-D-Grafik wissen sollte

Was immer wir tun, wo immer wir uns bewegen, haben wir es mit dreidimensionalen Gegenständen zu tun. Man möchte daher annehmen, nichts sei leichter, als solche Objekte zu beschreiben oder auch im Raum zu bewegen. Nun, es zeigt sich bald, daß dies gar nicht so einfach ist. Am liebsten würden wir sagen:

»Ich befinde mich in einer Zimmerecke und möchte vor mir eine braune Möbelgarnitur mit einer gelben Vase am Tisch mit vielen Blumen sehen. Das Licht kommt übrigens zum Fenster herein ...«

Das klingt zwar schon ganz konkret, aber ein bißchen mehr müssen wir dem Computer schon anvertrauen. Deswegen bleibt uns das folgende Kapitel nicht erspart.

3.1 Grundlegendes über die Perspektive

Gehen wir davon aus, daß unsere Objekte alle in einem Koordinatensystem beschrieben sind, so daß jeder Punkt durch seine Koordinaten (x, y, z) eindeutig festgelegt werden kann. Dann gilt dasselbe natürlich auch für den Standort des Beobachters bzw. für unsere fiktive Kameralinse. Kennen wir die Koordinaten dieses »Augpunktes« und vereinbaren wir, daß wir unser Objektiv vorläufig immer auf den Koordinatenursprung fixieren, dann wissen wir schon ziemlich viel. Was jetzt nur noch dazukommen könnte, ist, daß wir die Kamera verdrehen (weil z. B. der Hubschrauber, in dem wir fliegen, eine Kurve macht). Den Rest, also das Berechnen der zweidimensionalen Bildschirmkoordinaten, überlassen wir unserem Computer.

Halten wir fest: Durch Angabe der Koordinaten des Augpunktes und eines Verdrehungswinkels ist eine Perspektive bei Blickrichtung auf den Koordinatenursprung vollständig festgelegt. Deshalb werden unsere Datenfiles immer so beginnen, daß nach dem Namen der Szene (wegen der Übersichtlichkeit!) immer die Koordinaten des Augpunktes und (falls erwünscht) der Verdrehungswinkel angegeben werden.

```
NAME_DER_SZENE
augpunkt 20 30 10
verdrehung 45
```

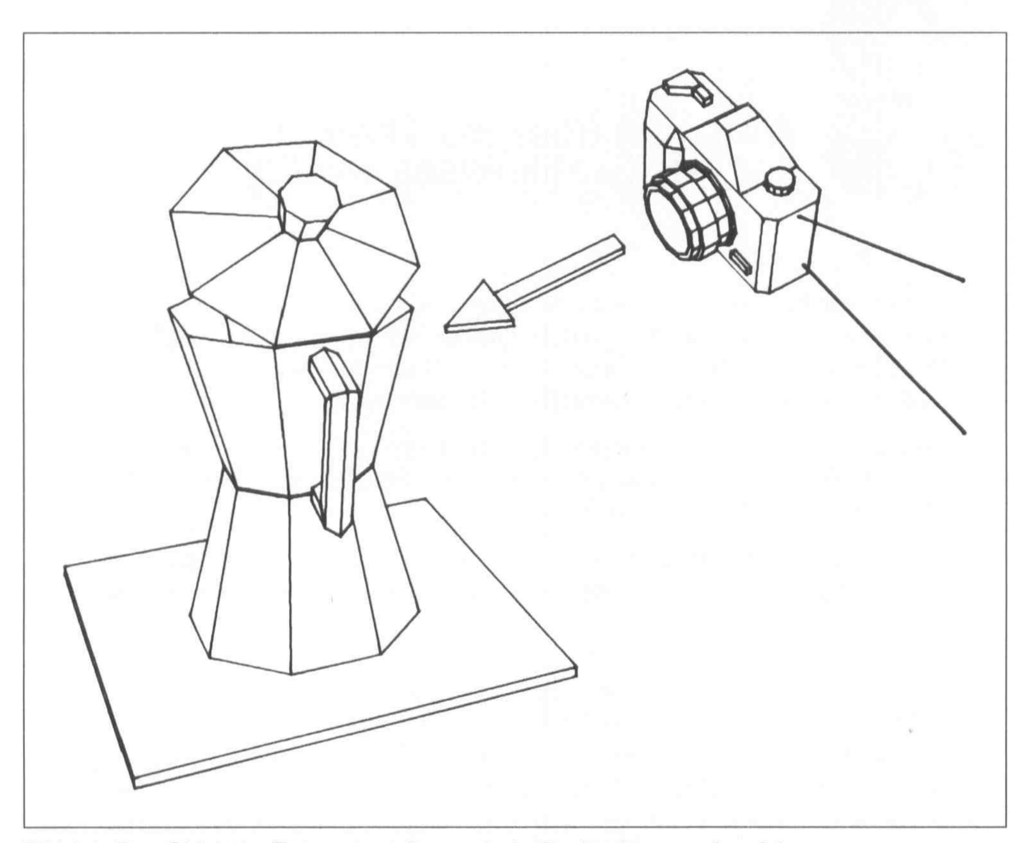


Bild 3.1: Das Objekt im Raum samt Augpunkt (= Beobachterstandpunkt)

Man beachte dabei, daß der Name der Szene zwar fast beliebig lang, nicht aber durch Leerzeichen getrennt sein darf. Wenn man aus optischen Gründen einen Abstand innerhalb des Namens lassen will, setzt man am besten einen Unterstrich (_).

Sowohl die Koordinaten des Augpunktes wie auch den Verdrehungswinkel kann man später von der Benutzeroberfläche verändern. Fehlt der Verdrehungswinkel im Datenfile, wird er automatisch mit 0 initialisiert.

Eine weitere Möglichkeit der Änderung der Perspektive ergibt sich, wenn wir die Achse unserer Linse nicht mehr auf den Koordinatenursprung richten wollen, sondern etwa auf den Punkt (3, –2, 4) oder das Zentrum der Szene. Dies erreichen wir durch eine zusätzliche (optionale) Datazeile

```
fixiere 3 -2 4 bzw. fixiere zentrum
```

Diese Wahlmöglichkeit gestattet es z. B., Objekte immer zu »fixieren«, auch wenn sie sich im Laufe einer Bewegung vom Koordinatenursprung entfernen (Bild 3.2). Schreiben wir keine solche Zeile ins Datafile, ist die Achse der Linse automatisch auf den Koordinatenursprung gerichtet.

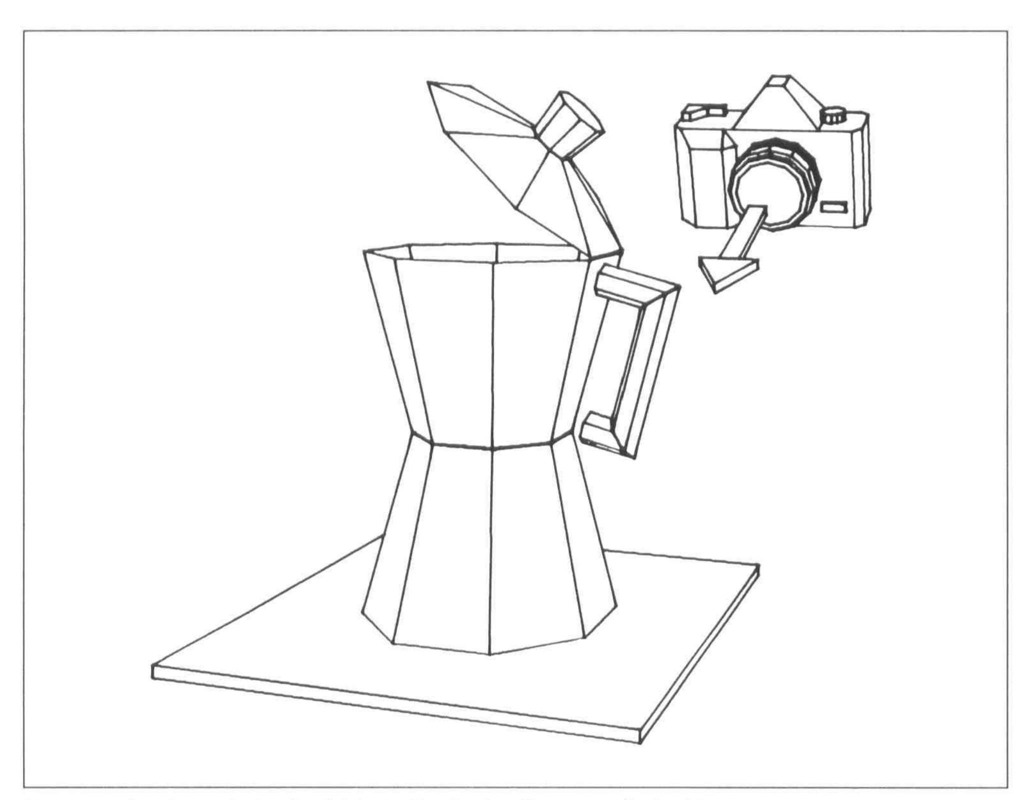


Bild 3.2: Man kann jeden beliebigen Punkt des Raums mit der Kamera anvisieren (»fixieren«)

Wie läßt sich nun die Brennweite unserer Kamera einstellen? Wir wissen ja, daß man beim Blick durch ein Weitwinkel-Objektiv einen ganz anderen Eindruck von der Szene erhält als beim Blick durch ein Teleobjektiv. Nun, die verschiedenen Brennweiten bei den Objektiven bewirken einfach einen anderen Bildausschnitt, der entsprechend vergrößert oder verkleinert dargestellt wird! Dementsprechend wollen wir uns merken:

Teleobjektiv = weit entferntes Auge + starke Vergrößerung

Weitwinkelobjektiv = weniger entferntes Auge + geringe Vergrößerung

Je weiter wir von den Objekten weggehen, desto kleiner wird das Bild. Vergrößern wir das Bild aber proportional zur Entfernung, erhalten wir Aufnahmen, die von einem Teleobjektiv stammen könnten (Bild 3.3). Umgekehrt können wir uns den Objekten nähern und gleichzeitig das Bild verkleinern, um eine »Fischaugenperspektive« zu erzeugen (Bild 3.4).

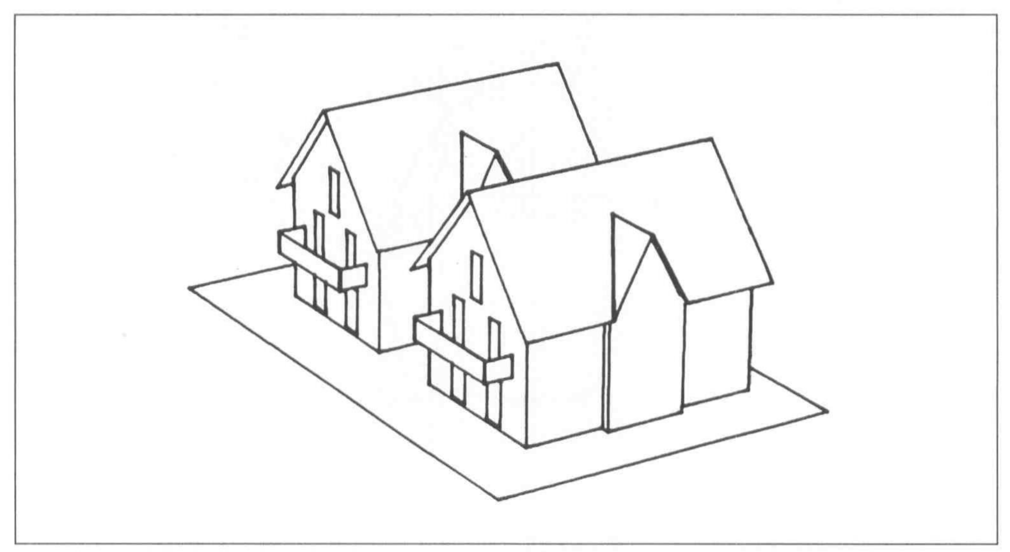


Bild 3.3: Teleobjekiv-Effekt: Der Augpunkt ist weit entfernt (Distanz größer als dreifacher Szenendurchmesser)

Das menschliche Auge sieht jenen Bildausschnitt scharf, den wir bei einem Objektiv mit etwa 60 mm Brennweite zur Verfügung haben (Bild 3.5). Dies entspricht der Forderung, daß der Durchmesser der Szene kleiner sein sollte als der durchschnittliche Abstand des Augpunktes von der Szene. Punkte außerhalb der Szene werden zwar von einer Kamera auch richtig abgebildet, erscheinen dem menschlichen Auge aber nur mehr unscharf. Wir empfinden daher solche Bilder als unnatürlich.

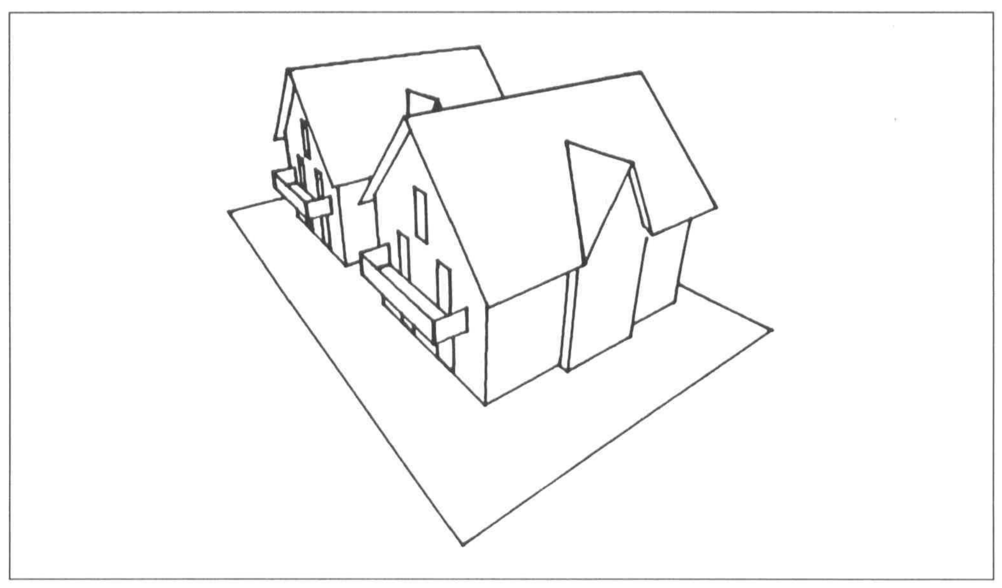


Bild 3.4: Fischaugen-Perspektive: Der Augpunkt ist nahe am Objekt
(Distanz kleiner als doppelter Szenendurchmesser)

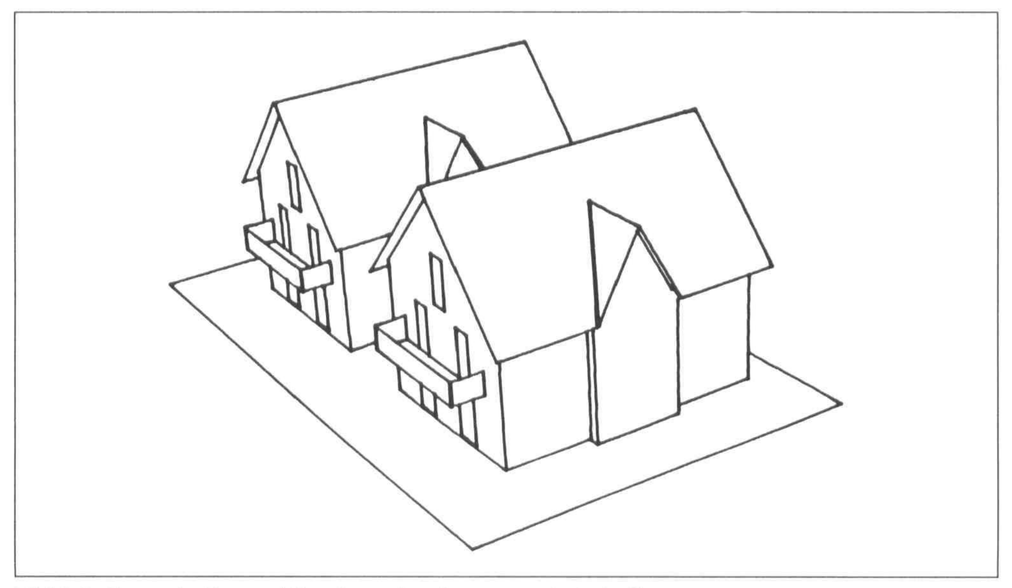


Bild 3.5: Normale Perspektive (so, wie wir es gewohnt sind):
Die Entfernung des Augpunktes vom Objektzentrum ist ca. doppelt so groß wie der Durchmesser der Szene. Dies entspricht einem Öffnungswinkel des »Sehkegels« von ca. 60 Grad

Was die Vergrößerung der Bildfigur auf der Zeichenfläche anbelangt, ist der 3-D-Sprinter darauf eingestellt, das Bild nicht zu groß oder zu klein zu zeichnen. Ist man mit dieser automatischen Vergrößerung bzw. Verkleinerung nicht einverstanden, kann man durch einen Maßstabsfaktor die Größe des Bildes manipulieren, indem man eine zusätzliche Datenzeile ins Datenfile schreibt. So bewirkt etwa die Datenzeile

```
vergroesserung 2
```

eine Vergrößerung des Bildes mit dem Faktor 2. Dies bedeutet, daß dem Auge das Objekt bereits viermal so groß erscheint, weil die Fläche der Bildfigur mit dem Quadrat des Längenmaßstabs ansteigt bzw. abnimmt. Im allgemeinen wird man mit einem Faktor zwischen 0.7 und 1.5 auskommen.

3.2 Wie postiert man die Kamera?

Die Wahl des Augpunktes ist von entscheidender Bedeutung dafür, daß die Bilder, die Sie mit dem 3-D-Sprinter erzeugen wollen, realistisch aussehen. Bitte lesen Sie daher dieses Kapitel aufmerksam durch und prägen Sie sich die kursiv dargestellten Merksätze ein.

Beginnen wir mit dem einfacheren Fall, daß Sie bereits ein Datenfile zur Verfügung haben, in dem auch Daten für die Perspektive eingetragen sind (wie wär's mit dem Datenfile Lampe?). Starten Sie das Programm interaktiv, warten Sie ein paar Bilder ab und drücken Sie auf die [Esc]-Taste (oder auch die [q]-Taste). Nun listen Sie die »nachricht«. Sie sehen, daß in den letzten Zeilen der »nachricht« die Daten für die Perspektive, die zuletzt verwendet wurde, angegeben sind: Augpunkt, Verdrehungswinkel, Vergrößerungsfaktor und eventuell auch der Punkt, auf den die Achse der Linse gerichtet war. Wenn Sie sich diese Werte in das Datenfile der Lampe eintragen, startet das Programm beim nächsten interaktiven Aufruf mit jenem Bild, das Sie vor dem letzten Abbruch gesehen haben.

Wenn Sie also eine möglichst günstige »Startposition« haben wollen, spielen Sie mit dem Datenfile eine Weile interaktiv herum, verdrehen die Objekte, zoomen sie heran usw., bis Sie endlich die Perspektive Ihrer Wahl gefunden haben. Dann kehren Sie mit [Esc] zur Benutzeroberfläche zurück und übertragen die Werte aus der »nachricht« in das jeweilige Datenfile.

Nehmen wir jetzt an, wir schreiben ein neues Datenfile: Wir wollen z. B. ein Objekt darstellen, von dem wir eine etwa 15 cm große Skizze gemacht haben. Der Einfachheit halber wählen wir nun 1 Einheit = 1 cm. Am besten nehmen wir prinzipiell den Nullpunkt des Koordinatensystems etwa im Zentrum des Objekts an. Damit werden alle Eckpunkte Koordinaten haben, die im Bereich zwischen – 8 cm und + 8 cm liegen.

Das Auge sollte nun auf jeden Fall weiter entfernt sein als der Durchmesser des Objekts, also sagen wir 20 bis 30 cm. Dann wären Augpunktkoordinaten wie etwa (15, 10, 15) oder (–18, – 4, 10) akzeptabel. Warum? Nun, der Abstand des Punktes (x, y, z) vom Ursprung ist nach dem »räumlichen Pythagoras« gleich:

$$\text{Distanz} = \text{Wurzel}\ (x^2 + y^2 + z^2),$$

bei den beiden vorgeschlagenen Punkten also ca. 24 cm bzw. 21 cm. Natürlich wird niemand die angegebene Formel wirklich verwenden, aber es ist gut, sie im Kopf zu haben.

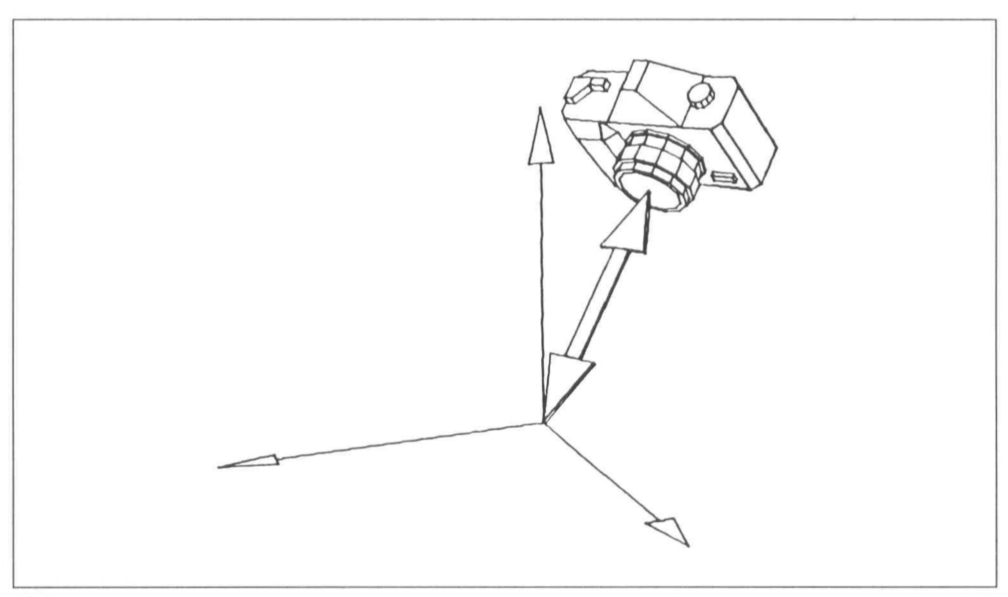


Bild 3.6: Die Distanz des Augpunktes

Merken wir uns die folgende Faustregel:

Es sollen nicht alle Augpunktkoordinaten kleiner als der Durchmesser der Szene sein (das Vorzeichen der Koordinaten bleibt bei dieser Regel außer acht).

Die z-Koordinate des Augpunktes enscheidet darüber, ob wir es mit einer Obersicht (z-positiv) oder einer Untersicht zu tun haben. Eine Ansicht von oben erreichen wir, wenn die x- und die y-Koordinate des Augpunktes verschwindet. Analog haben wir es mit einer Ansicht von vorne zu tun, wenn die y- und z-Koordinate des Augpunktes gleich Null sind usw.

Übrigens: Durch Drücken der Tasten 1 bis 6 (aber auf der gewöhnlichen Tastatur und nicht am Zahlenblock!) erreichen Sie der Reihe nach eine Ansicht von oben, vorne, rechts, unten, hinten, links (dabei wird allerdings der Augpunkt sehr weit weggerückt, so daß man von einer »Normalprojektion« spricht).

Der 3-D-Sprinter vergleicht die Distanz des Augpunktes mit dem maximalen Durchmesser der Szene. Sind wir doch mit dem Auge zu nahe dran, bricht er mit einer entsprechenden Meldung ab. Wenn wir im Verlauf einer Animation durch wiederholtes Drücken der [Shift]- +[d]-Taste an die Objekte herangehen, tritt ab einer gewissen Minimalentfernung eine Sicherheitssperre in Kraft, die eine weitere Annäherung an die Szene verhindert. Der Grund dafür liegt nicht zuletzt darin, daß die Hardware des Amiga zum Absturz kommen kann, wenn eine Seitenfläche unseres Objekts so groß wird, daß sie die Grenzen des Bildschirms an drei Stellen überschreitet.

Weniger kritisch ist die Wahl zu großer Augpunktkoordinaten. Sie bewirken nur etwas abgeflachte und nicht so plastische Bilder. Die Hobby-Fotografen unter uns machen sich diese abflachende Wirkung oft zunutze, wenn sie Porträtaufnahmen mit Teleobjektiven machen, um vorteilhafte Bilder zu erhalten. Wer will schon auf seinem Paßbild eine besonders dreidimensional wirkende Nase haben?

Merken wir uns also zusätzlich:

Keine der Augpunktkoordinaten sollte das Dreifache des Durchmessers der Szene überschreiten.

Der 3-D-Sprinter beschwert sich allerdings nicht über zu große Distanzen. Wenn wir nämlich eine technische Zeichnung am Plotter ausgeben wollen, sind perspektivische Verzerrungen oft unerwünscht. Ein Techniker kann aus einer sogenannten »Normalprojektion« wesentlich mehr Informationen ablesen als aus einer Perspektive (bei speziellen Ansichten sind viele Werte sogar in wahrer Länge abzunehmen).

Bei technischen Plänen wählt man oft sehr große Augpunktkoordinaten (z. B. (1000, –2000, 1500)), um eine Normalprojektion zu erreichen. Insbesondere liefert (10000, 0, 0) einen Aufriß, (0, 10000, 0) einen Kreuzriß und (0, 0, 10000) einen Grundriß (Bild 3.7).

Tippen wir einfach einmal unsere Augpunktkoordinaten und irgendwelche Phantasiewerte für das Lichtzentrum nach dem Namen des Datenfiles ein:

```
TESTFILE
augpunkt 30 15 25
licht 100 100 100
```

Starten Sie den 3-D-Sprinter interaktiv. Wenn alles gutgegangen ist, erscheint ein perspektivisches Bild der beschriebenen Objekte am Bildschirm, bereits einigermaßen gut in den Bildschirm eingepaßt. Die optimalen Augpunktkoordinaten holen wir uns wie beschrieben durch interaktives Herumspielen. Durch Angabe eines Vergrößerungsfaktors kann die Größe des Bildes noch optimiert werden.

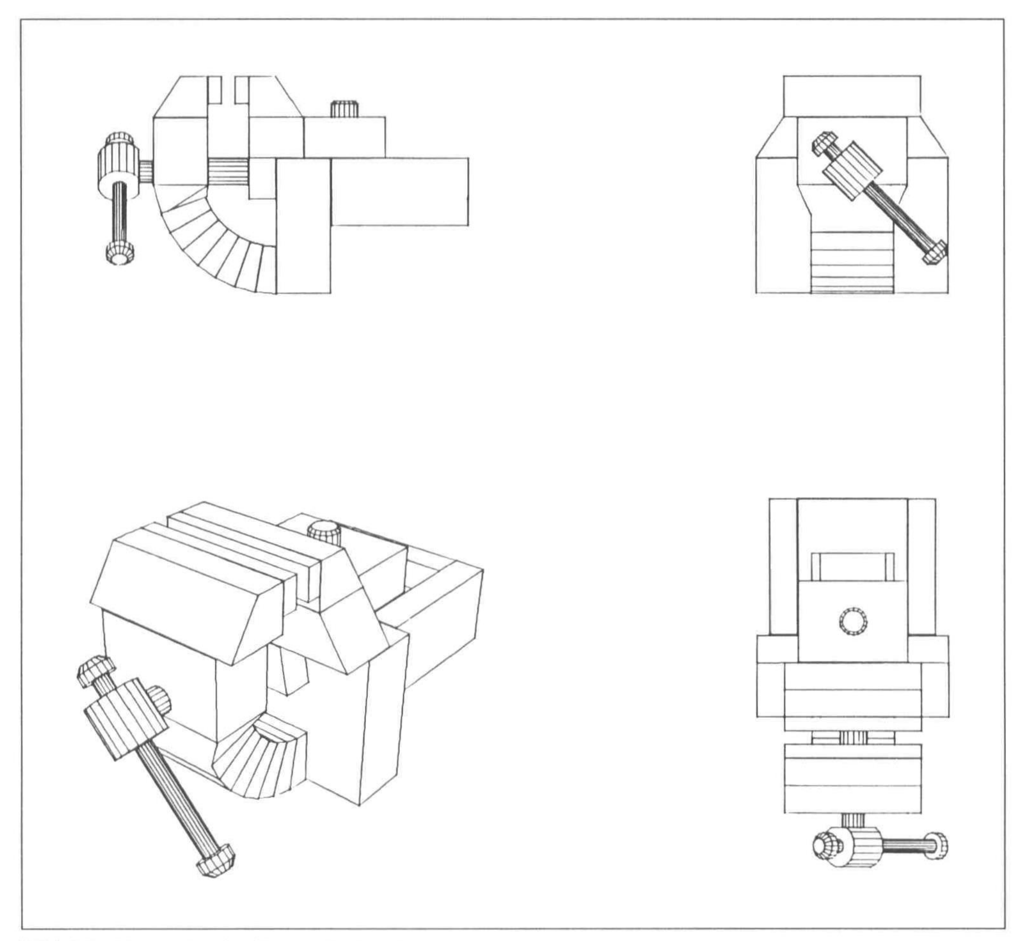


Bild 3.7: Normalprojektionen bei technischen Objekten
Spezielle Ansichten: (Datenfile »schraubstock«)
a) von oben (Grundriß),
b) von vorne (Aufriß),
c) von rechts (Kreuzriß),
d) allgemeine Normalprojektion (normale Axonometrie)

Zur Festlegung gut geeigneter Augpunktkoordinaten gibt es eine zweite, etwas mühsamere Methode: Wählen Sie dazu in der Benutzeroberfläche den Menüpunkt »Edit->Augpunkt« an, und klicken Sie der Reihe nach in die Felder für die x-, y- bzw. z-Koordinaten des Auges, um danach neue Zahlenwerte einzutippen. Beim neuerlichen Aufruf des Datenfiles von der Benutzeroberfläche werden die geänderten Augpunktkoordinaten benutzt. Vergessen Sie nicht, das geänderte Datenfile vor dem Verlassen des 3-D-Sprinters zu speichern!

3.3 Wo postiert man die Lichtquelle?

Licht-Schatten-Wirkungen sind das Salz in der Suppe einer 3-D-Zeichnung. Die Wahl der Lichtquelle trägt somit ebenfalls entscheidend zur Bildqualität bei. Der 3-D-Sprinter erlaubt es, die Lichtquelle an jeder beliebigen Stelle des Raumes zu setzen, die nicht innerhalb eines Objekts ist. Trotzdem empfiehlt es sich, gewisse Regeln einzuhalten:

Wir sind von Natur aus das Sonnenlicht am ehesten gewöhnt, also eine Lichtquelle, die weit entfernt ist. Wenn wir nicht unbedingt spezielle Effekte erreichen wollen, wählen wir die Koordinaten des Lichtpunktes groß (der Abstand des Lichtzentrums sollte mehrere Objektdurchmesser betragen). Ebenso sind wir eher Licht gewöhnt, das von oben kommt. Im allgemeinen wird daher die z-Koordinate des Lichts positiv sein.

Machen wir ein konkretes Beispiel: Wir sehen uns mit einem Editor das Datenfile »3-D-Sprinter:datas/photo« an. Falls Sie mit einer Kopie der Originalversion des 3-D-Sprinters arbeiten, sollten die ersten Zeilen des Datenfiles so aussehen:

```
kPHOTO_APPARAT
augpunkt 10 20 20
licht 150 -50 100
vergroesserung 1.3
```

Sehen wir uns die Szene einmal an, indem wir das Programm interaktiv starten und gleich nach Eröffnen des Zeichenbildschirms auf die [f]-Taste drücken (f = freeze verhindert eine Bewegung des Bildes). Drücken wir zusätzlich die [s]-Taste (s = shade), dann erscheint Sekunden später der Fotoapparat schattiert am Bildschirm. Beim Drücken der [c]-Taste (c = cast shadows) werden zusätzlich auch die Schlagschatten eingezeichnet, was ebenfalls nur wenige Sekunden dauert (schließlich heißt Ihr Programmpaket nicht umsonst 3-D-Sprinter!).

Die Wahl der Lichtquelle ist schon ganz gut getroffen, finden Sie, aber Sie wollen trotzdem eine andere Beleuchtung ausprobieren. Na gut, drücken Sie die [l]-Taste (l = light). Wenn das Bild, das der 3-D-Sprinter im Hintergrund für Sie gerade aufbaut, fertiggezeichnet ist, erscheinen am Bildschirm die Koordinatenachsen und ein Quader, dessen Raumdiagonale zur Lichtquelle weist.

Nun manipulieren Sie interaktiv die Lichtrichtung mit den Cursortasten auf dem Zahlenblock: Aufwärtspfeil für Lichtquelle höher, Abwärtspfeil für Lichtquelle tiefer, Pfeil nach links für Rotation der Lichtquelle um die z-Achse im Uhrzeigersinn, Pfeil nach rechts für Rotation im Gegensinn (von oben gesehen). Damit dies interaktiv geschehen kann, wird das Bild der Szene nicht jedesmal neu berechnet, so daß Sie die Änderung der Schattierungsverhältnisse nicht sofort sehen können. Wenn Sie glauben, die neue Lichtrichtung könnte günstig sein, dann verlassen Sie den momentanen Modus durch einmaliges Drücken der [ESC]-Taste oder [q]-Taste. Jetzt bereitet der 3-D-Sprinter im Hintergrund das Bild bei geänderten Lichtverhält-

nissen vor und wird es Ihnen in wenigen Sekunden zeigen. Wenn Ihnen das Bild nicht so gut gefällt, wiederholen Sie den Vorgang mit der [l]-Taste. Sind Sie geradezu begeistert von der Wahl der neuen Lichtquelle, dann steigen Sie aus dem Zeichenvorgang mit [Esc] oder [q] aus. Dies bewirkt wieder, daß der 3-D-Sprinter die aktuellen Daten, insbesondere auch die neugewonnenen Lichtpunktkoordinaten, in Ihre »nachricht« schreibt. Sehen Sie das »nachricht«-File an. Dort sollte in einer der letzten Zeilen eine Mitteilung stehen, in der Ihre neugewählten Lichtkoordinaten eingetragen sind, also etwa:

```
licht 152.10 -27.31 189.52
```

Jetzt liegt es an Ihnen, diese Koordinaten in das Datenfile einzutragen oder nicht. Wenn Sie es tun, initialisiert der 3-D-Sprinter von nun an die Lichtquelle an der neu von Ihnen gewählten Stelle. Vergessen Sie nicht, die Änderung auch abzuspeichern!

Wie bei der Festlegung des Augpunktes gibt es eine zweite, etwas mühsamere Methode zur Ermittlung gut geeigneter Lichtkoordinaten. Sie besteht darin, in der Benutzeroberfläche den Menüpunkt Edit->Lichter->1 anzuwählen und nach Anklicken der x-, y- bzw. z-Koordinate des Lichts diese Koordinaten zu ändern. Beim neuerlichen Aufruf des Datenfiles von der Benutzeroberfläche werden die neugewählten Lichtkoordinaten als Ausgangswerte verwendet. Beim längeren Betrachten einer bewegten Szene mit eingeschalteten Schlagschatten fällt Ihnen sicher auf, daß die Lichtquelle immer festzubleiben scheint, während sich die Szene bewegt. Dadurch entstehen ständig neue Schattenwirkungen. Ein Beobachter, der sich mit der Szene mitbewegen würde, hätte dabei umgekehrt den Eindruck, daß sich die Lichtquelle bewegt, während seine Umgebung starr zu sein scheint. Dies ist derselbe Effekt, den wir Erdenbürger tagtäglich beobachten, wenn sich die Sonne über den Horizont »bewegt«. In Wirklichkeit wissen spätestens seit dem berühmten Ausspruch von Galilei alle, daß wir es sind, die sich relativ zur Sonne bewegen.

Der 3-D-Sprinter ist im interaktiven Modus von vornherein »heliozentrisch« eingestellt, das heißt, wenn wir ihm nicht durch eine Datenzeile nach den Koordinaten des Lichtpunktes sagen, daß wir es anders wollen, bleibt die Lichtquelle relativ zum Bildschirm konstant, so daß die Objekte ständig anders beleuchtet werden. Die Datenzeile nach den Lichtkoordinaten, die das Licht »geozentrisch« bzw. »heliozentrisch« macht, heißt dementsprechend

`geozentrisch` bzw. `heliozentrisch`

Wird keine solche Zeile angegeben, wählt das Programm im interaktiven Modus automatisch heliozentrisch. Beim Abarbeiten von Animations-Skripten (Kapitel 5) ist der 3-D-Sprinter allerdings von vornherein geozentrisch eingestellt. Wenn wir nämlich z. B. einen Hubschrauberrundflug um ein Gebäude machen, wird sich der Sonnenstand in dieser kurzen Zeit nur unmerklich verändern.

3.4 Verschiedene Darstellungsarten

Wir haben schon so oft von verschiedenen Darstellungsarten gesprochen, daß wir diese einmal übersichtlich zusammenfassen wollen: Wenn wir nicht über die Benutzeroberfläche oder ein Animations-Skriptum eine andere Darstellungsart gewählt haben, zeichnet der 3-D-Sprinter vorerst nur ein **Drahtgittermodell** unserer Szene (englisch: wireframe, daher die Taste [w], um diesen Modus zu aktivieren). Der große Vorteil des Drahtgittermodells ist, daß selbst kompliziertere Szenen ohne größeren Zeitverlust interaktiv bewegt werden können. Um die räumliche Vorstellung zu unterstützen, werden Punkte, die sich weiter vom Auge entfernt befinden, dunkler dargestellt als Punkte im Vordergrund. Dieses Verfahren heißt im Englischen »depth-cueing« und ist eine enorme Visualisierungshilfe.

Die schnellste Art der anschaulichen Darstellung bei Unterdrückung unsichtbarer Objektteile ist die **schattierte Ausgabe** (englisch: shading, daher die [s]-Taste, um diese Darstellungsart zu aktivieren). Die Bezeichnung ist ein wenig irreführend, weil sie den Eindruck erweckt, daß wirkliche Schatten eingezeichnet sind. Unter einer schattierten Darstellung versteht man aber nur das Einfärben der Objektfacetten mit Farben aus einer Farbpalette. Der Neigungswinkel der Facette zum Lichtstrahl ist der Hauptfaktor bei der Berechnung des Helligkeitswertes. Hinzu kommen noch Abstand vom Augpunkt und vom Lichtzentrum.

Die große Stärke des 3-D-Sprinters aber ist die **Darstellung mit Schlagschatten** (englisch: cast shadows, daher die [c]-Taste zum Aktivieren dieses Modus). Objekte werfen ja Schatten auf andere Objekte und unter Umständen auf sich selbst. Bilder mit korrekten Schatten waren bisher den sogenannten »Raytracing-Programmen« vorbehalten, wobei man mit extrem langen Rechenzeiten rechnen mußte. Doch Genaueres dazu im nächsten Abschnitt.

Eine vierte Darstellungsart ist das **Drahtgittermodell mit Unterdrückung unsichtbarer Linien bzw. Linienteile** (englisch: hidden lines, daher die [h]-Taste zur Aktivierung), die vor allem für Plotterausdrucke oder Hardcopies vom Bildschirm geeignet ist. Alle Schwarzweiß-Bilder in diesem Buch wurden mit diesem Modus erzeugt.

Auch diese Darstellungsart ist von der Bildschirmgröße unabhängig und erzeugt exakte Bilder, worunter die Rechenzeit ein wenig leidet. Zur zusätzlichen Unterstützung der Vorstellungskraft werden zumindest am Bildschirm weiter hinten liegende Streckenteile wieder dunkler dargestellt als Strecken im Vordergrund.

3.5 Warum ist der 3-D-Sprinter so schnell?

Denken Sie an einen Würfel. Ein Würfel ist ein sogenannter konvexer Körper, das heißt grob gesprochen, es handelt sich dabei um einen Vollkörper ohne Einbuchtungen und Einschnürungen. Konvexe Körper haben Eigenschaften, die man in der Computergeometrie besonders gewinnbringend anwenden kann. Insbesondere werfen sie auch keine Schatten auf sich selbst, und der Schatten auf andere Objekte kann schneller berechnet werden. Weil der 3-D-Sprinter die Eigenschaften konvexer Körper optimal ausnützt, eignet er sich besonders gut zur Darstellung von Szenen, die aus konvexen Bausteinen zusammengesetzt sind. Und damit läßt sich schon was machen, wie man im folgenden Bild sehen kann.

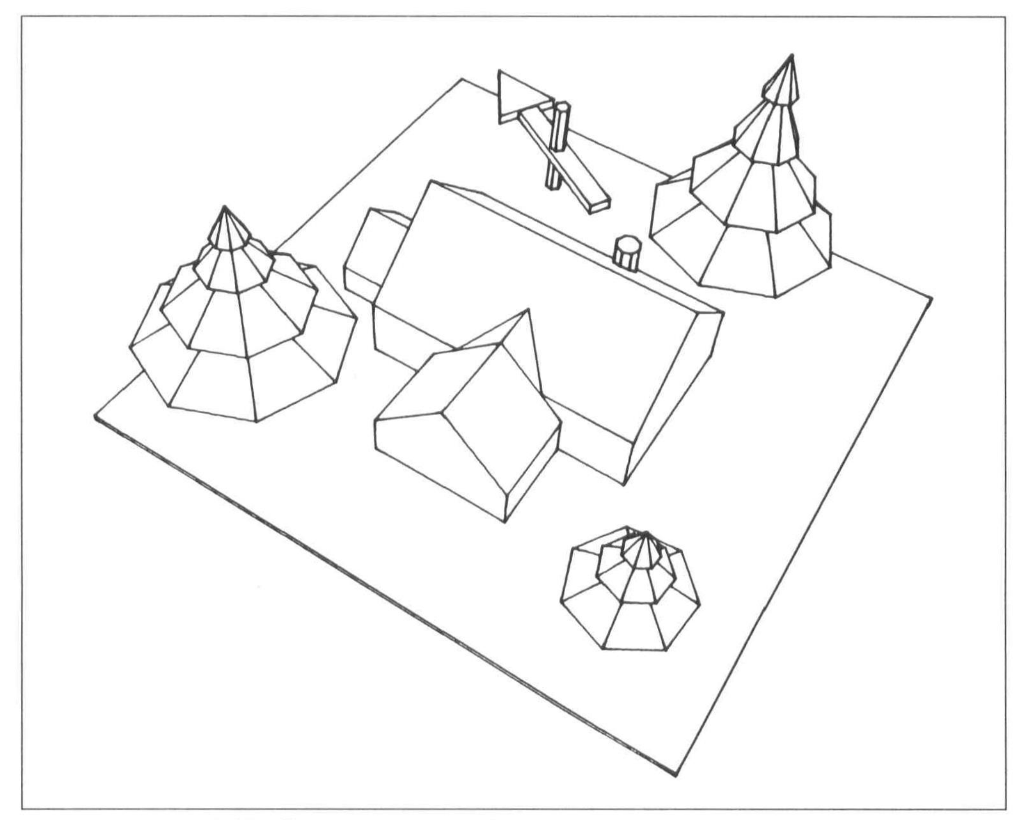


Bild 3.8: Ein Beispiel für Zusammensetzung konvexer Körper
(Datenfile »Haeuschen_im_Wald«)

Der 3-D-Sprinter benutzt aber auch Eigenschaften anderer Objekttypen. Es sind dies vor allem die Drehflächen, auch wenn sie nicht konvex sind, gefolgt von den Funktionsgraphen und Objekten, die in Schichten aufgebaut sind. Immer werden gewisse Regelmäßigkeiten der Objekte dazu verwendet, die Reihenfolge beim Zeichnen der einzelnen Facetten schnell zu bestimmen. Schauen Sie dem 3-D-Sprinter einmal beim Zeichnen zu (das erledigt er normalerweise im Hintergrund), indem Sie die [z]-Taste drücken (wie wär's mit »z wie zuschauen«, nachdem die [f]-Taste (»foreground«) schon für freeze und die [w]-Taste (»watch drawing«) schon für wireframe vergeben sind). Sie sehen, wie das Programm die Flächen in solcher Reihenfolge zeichnet, daß durch das gegenseitige Übermalen die Sichtbarkeit am Ende stimmt.

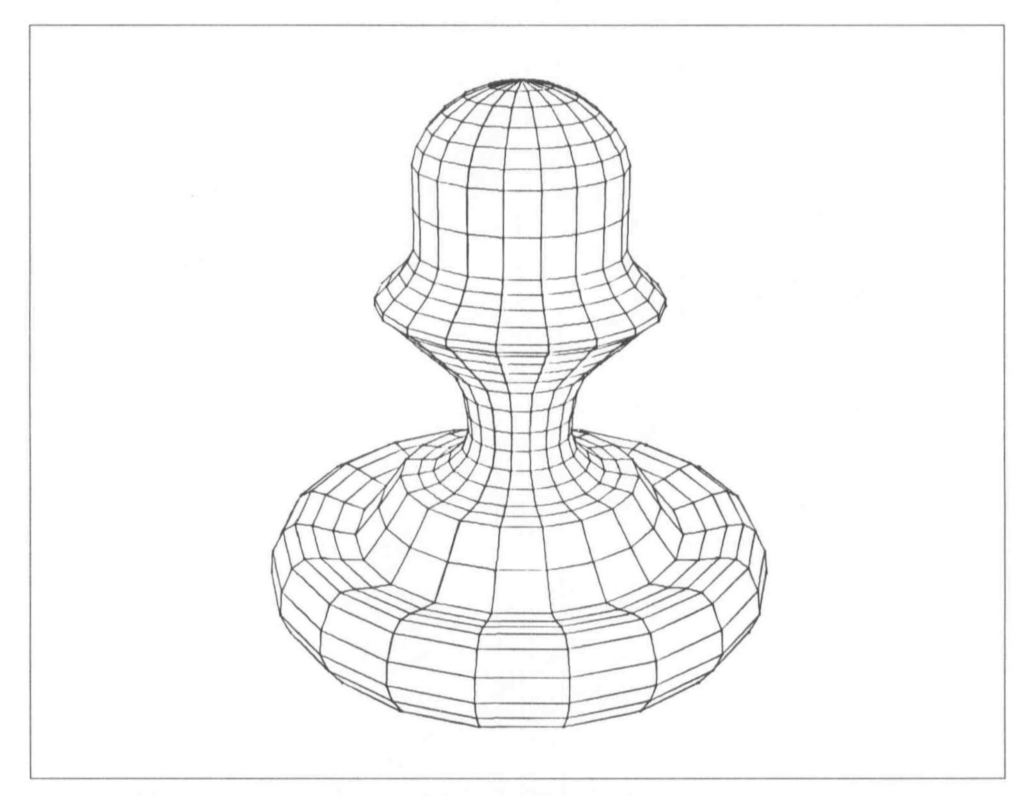
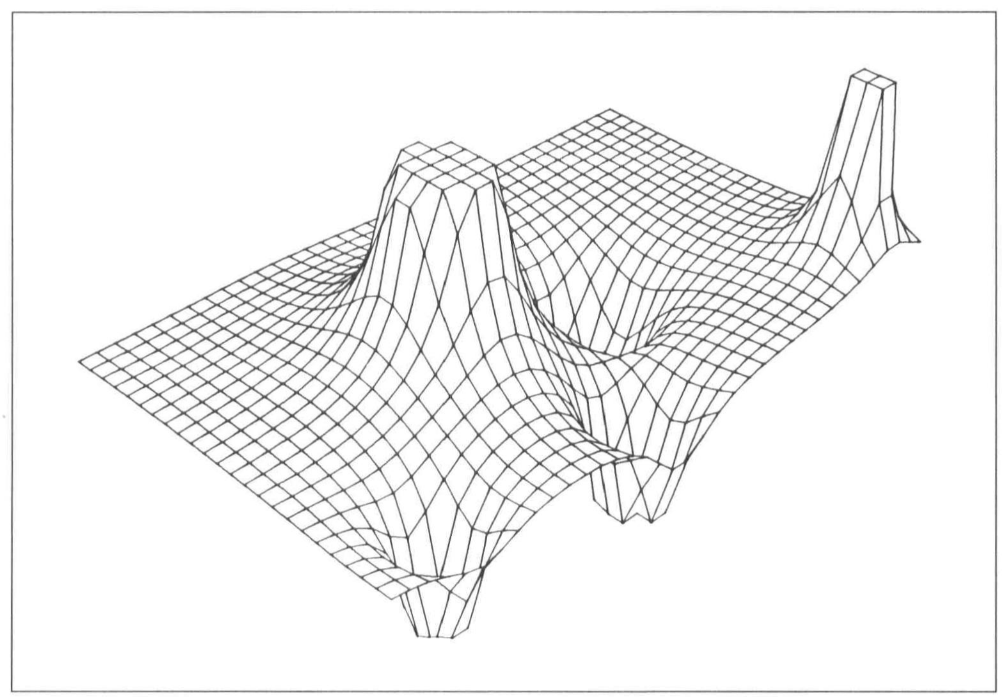


Bild 3.9: Spezielle Objekttypen: a) nichtkonvexe Drehfläche



b) Funktionsgraph

Durch die Anwendung neuartiger Berechnungsmethoden schafft der 3-D-Sprinter Bilder von ansprechender Qualität in unvergleichlich kürzerer Rechenzeit als etwa das Raytracing-Verfahren. Zusätzlich »explodieren« die Rechenzeiten bei den Raytracing-Programmen bei zunehmender Bildschirmauflösung, während der 3-D-Sprinter unabhängig von der Auflösung arbeitet!

Doch wo viele Vorteile sind, muß auch ein Nachteil sein. Der besteht beim 3-D-Sprinter darin, daß man sich an gewisse Spielregeln halten muß, wenn man wirklich korrekte Ergebnisse erhalten will. Deshalb empfehlen wir Ihnen dringend, den nächsten Teilabschnitt genau durchzulesen (auf ihn werden Sie öfter zurückkommen, wenn Sie eigene Datenfiles zu schreiben beginnen).

3.6 Die wichtigsten Bausteintypen

Im Datenfile, das der 3-D-Sprinter einliest, kommen immer wieder Schlüsselwörter vor, die vom Programm entsprechend verarbeitet werden. Sehen wir uns einmal systematisch an, was denn da eingegeben werden kann oder auch muß: Den Kopf des Datenfiles bilden die Datenzeilen zur Bestimmung der Perspektive und des Lichts.

Dann folgt der Name des ersten Bausteins. Dies dient hauptsächlich dazu, daß die Datenfiles leserlich bleiben. Der Name muß wie immer ohne Leerzeichen geschrieben werden. Dann kommt die Farbe des Bausteins (bei der Standardpalette vorerst nur Grau, Grün oder Gelb). Wenn wir eine andere Farbe hinschreiben (sagen wir Grün mit [ü]), nimmt der 3-D-Sprinter automatisch Grau und schreibt »unbekannte farbe grün (grau genommen)«. Jetzt muß der Typus des Bausteins folgen. Hier eine Auflistung aller bisher im Programm eingeflochtenen eingebauten Bausteintypen:

3.6.1 Der Typ »quader«

Das einfachste räumliche Gebilde ist wohl ein Würfel oder allgemeiner ein Quader. Wenn Länge, Breite und Höhe des Quaders bekannt sind, ist das Objekt bestimmt. Also, warum schreiben wir nicht einfach

```
MEIN_ERSTER_QUADER gelb quader voll 3 2 1
```

ins Datenfile und schauen, was rauskommt? Das Wörtchen voll übrigens bedeutet, daß der Körper massiv und nicht etwa hohl ist.

Starten wir den 3-D-Sprinter. Tatsächlich, das Programm zeichnet das Drahtgittermodell eines gelben Quaders, wobei sich die Kantenlängen wie 3 : 2 : 1 verhalten. Drücken wir die [s]-Taste: Ja, der Quader ist ein Vollkörper. Jetzt zurück in den Editor. Vertauschen wir voll durch hohl. Jetzt müßte der Quader hohl sein, wobei der Boden und der Deckel der Schachtel weggelassen sind. Schreiben wir nun eine weitere Datenzeile:

```
MEIN_ZWEITER_QUADER gruen wie MEIN_ERSTER_QUADER
      translation z 2
```

Dies bewirkt, daß ein zweiter (grüner) Baustein angelegt wird, wobei die Daten des ersten einfach übernommen werden, zu den Punktkoordinaten aber immer der z-Wert 2 addiert wird. Die beiden Quader befinden sich daher übereinander. Das Wörtchen »wie« macht's möglich (Bild 3.10).

Das zweite neue Wort ist »translation«, also Verschiebung. Es bewirkt, daß der Quader im Raum herumgeschoben werden kann. Dabei gelangt der Eckpunkt mit den kleinsten x- , y-, z-Koordinaten, dem ohne dieses Schlüsselwort die Koordinaten (0, 0, 0) zugeordnet werden, an die Stelle, die der Schubvektor angibt. Sie können je nach Geschmack

```
translation 3 4 -5     /* Eckpunkt gelangt nach (3,4,-5) */
translation x -3       /* entspricht translation -3 0 0  */
translation y  6       /* entspricht translation  0 6 0  */
```

schreiben. Eine weitere Feinheit: Sie dürfen und sollen in Ihr Datenfile auch Kommentare einsetzen, die dann (wie in der Programmiersprache »C«, in der das gesamte Programmpaket geschrieben wurde) durch die Zeichenfolge /* eingeleitet und mit der Zeichenfolge */ abgeschlossen werden. Kommentare dürfen auch »verschachtelt« sein, also etwa so:

```
/* Dies ist ein Kommentar:
   /* dies ist ein verschachtelter Kommentar */
   nun beenden wir den ursprünglichen Kommentar
*/
```

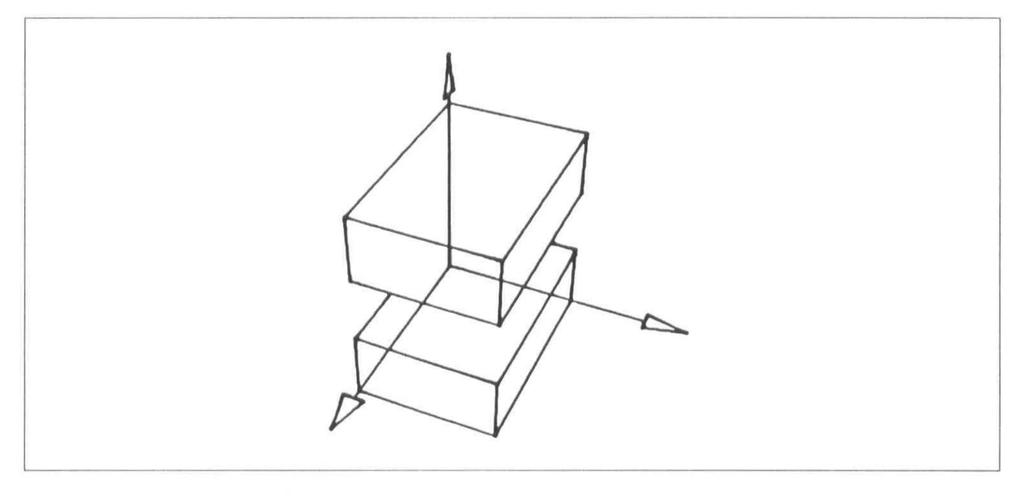


Bild 3.10: Quader als einfachste Bauelemente

Wenn Sie wollen, daß der hohle Quader unten geschlossen ist, erreichen Sie dies, indem Sie die Datenzeile

```
SCHACHTEL_MIT_BODEN grau quader hohl 2 3 5 +boden
```

eintragen. Nach dem Wörtchen »hohl« können Sie die Schlüsselwörter +boden bzw. +deckel anhängen.

3.6.2 Der Typ »drehflaeche«

In jedem halbwegs brauchbaren 3-D-CAD-Paket muß es einigermaßen einfach sein, Drehflächen zu erzeugen. Dies deshalb, weil in unserer Umgebung sehr viele Gegenstände »rund« sind. Der einfachste Rundkörper ist ein Drehzylinder. Einen Drehzylinder mit der z-Achse als Zylinderachse können wir erzeugen, wenn wir zwei Raumpunkte, die auf einer lotrechten Geraden liegen, um die z-Achse rotieren lassen (Bild 3.11). Die Punkte auf der lotrechten Geraden bezeichnet man als »Meridianpunkte«.

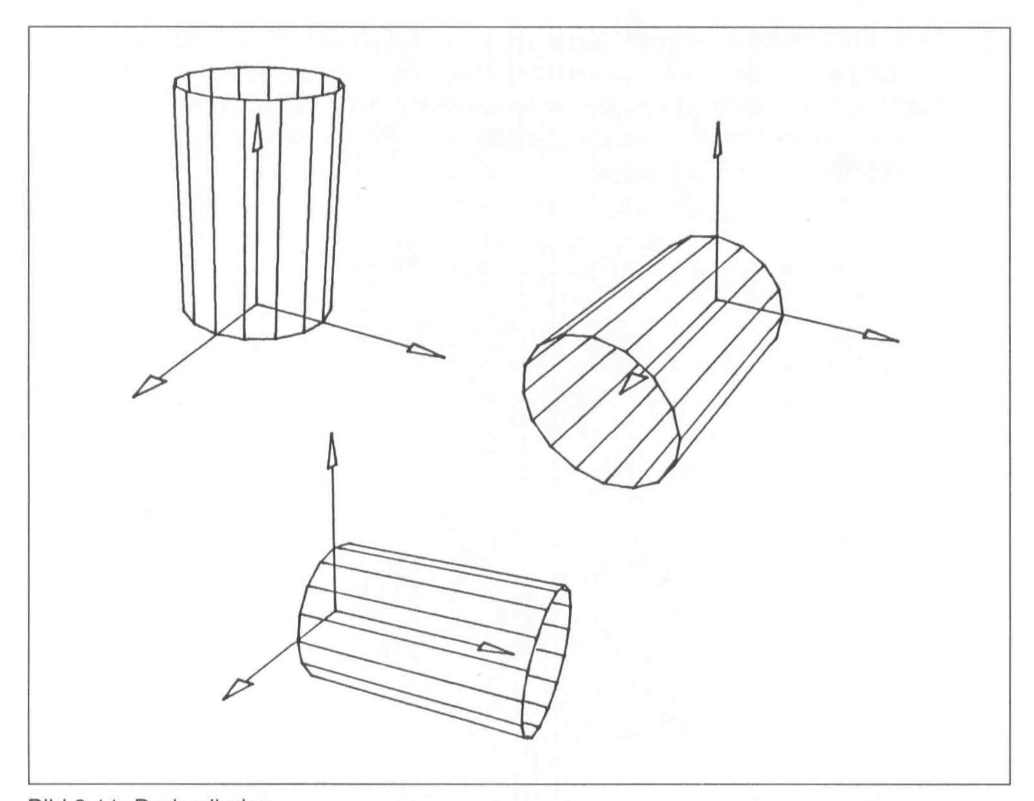


Bild 3.11: Drehzylinder
a) in »Hauptlage«
b) um 90 Grad um die y-Achse gedreht
c) um 90 Grad um die x-Achse gedreht

Dementsprechend sollten die Datenzeilen

```
MEIN_ERSTER_DREHZYLINDER gruen drehflaeche voll
   meridian 3 0 0, 3 0 5 rot_zahl 16
```

einen grünen Zylinder der Höhe 5 mit dem Radius 3 erzeugen, der auf der Basisebene z = 0 steht. Eigentlich handelt es sich dabei um ein regelmäßiges 16seitiges Prisma, das aber schon beinahe wie ein Drehzylinder aussieht. Durch Erhöhen der Rotationszahl wird der zylinderähnliche Eindruck noch verstärkt. Die Obergrenze für die Rotationszahl ist allerdings mit Rücksicht auf Rechenzeit und Speicherplatz mit 50 angesetzt. Ein 50seitiges Prisma kann man mit freiem Auge kaum mehr von einem Zylinder unterscheiden. Die höchste Rotationszahl, die bei Bildern in diesem Buch verwendet wurde, ist übrigens 30! Durch eine Zusatzzeile wie z. B.

```
translation 4 -3 7  /* Addiert den Schubvektor 4 -3 7 */
```

kann der Zylinder im Raum hin- und hergeschoben werden, wobei seine Achse stets lotrecht bleibt. Was aber, wenn der Zylinder geneigt sein soll? Bleiben wir bei unserem Beispiel: Wenn wir die obigen Datenzeilen durch die Zeile

```
rotation x 90  /* Rotation um die x-Achse um 90 Grad */
```

ergänzen, dann wird die y-Achse zur Zylinderachse (Bild 3.11). Analog macht

```
rotation y 90
```

die x-Achse zur Zylinderachse (Bild 3.11).

Der 3-D-Sprinter erlaubt Ihnen nun, beliebig oft irgendwelche Translationen und Rotationen hintereinander auszuführen. Dadurch sollte es für Sie ein leichtes sein, jede gewünschte Stellung des Zylinders zu erreichen. Dabei kann man recht gut sein räumliches Vorstellungsvermögen schulen.

Beachten Sie bitte, daß es nicht dasselbe sein kann, wenn man einen Körper zuerst irgendwohin verschiebt und dann um eine Koordinatenachse dreht oder ihn zuerst dreht und dann verschiebt!

Doch bevor wir noch genauer auf das Kapitel Drehungen und Rotationen eingehen (Kap. 7.2), wollen wir sehen, wie leicht es ist, auch andere rotationssymmetrische Körper zu erzeugen.

Nehmen wir uns einmal die Pyramiden bzw. Drehkegel vor: Ersetzen wir bei unserem Drehzylinder einen der Meridianpunkte durch einen Punkt auf der z-Achse, so entsteht bei Rotation dieser Geraden eine Pyramide bzw. ein Kegel, wenn die Rotationszahl groß genug ist, sagen wir 20 (Bild 3.12). Nehmen wir die Meridianpunkte so an, daß ihre Trägergerade die Achse trifft, entsteht ein Pyramidenstumpf bzw. Kegelstumpf. Der kann, ganz wie bei den Quadern und den Prismen (Zylindern) auch hohl sein (Bild 3.12):

```
PYRAMIDENSTUMPF grau drehflaeche hohl
    meridian 3 3 0, 2 2 2    /* Basisseite 6, Höhe 2 */
    rot_zahl 4               /* vierseitig */
```

Vorsicht: Wenn die Trägergerade der Meridianpunkte die z-Achse nicht schneidet, ist das Ergebnis kein Pyramidenstumpf (Kegelstumpf), sondern ein »windschiefes« Gebilde. Dies hat zur Folge, daß der 3-D-Sprinter u. U. eine falsche Sichtbarkeit ermittelt. In einem solchen Fall wird eine Warnung in das »Nachrichten-File« geschrieben. Ein Abbruch ist deshalb nicht sinnvoll, weil man, wie wir im Farbbild 14 (»Fußball-Pokal«) sehen können, durchaus nette Effekte mit Meridianpunkten erreichen kann, die nicht in einer Meridianebene liegen.

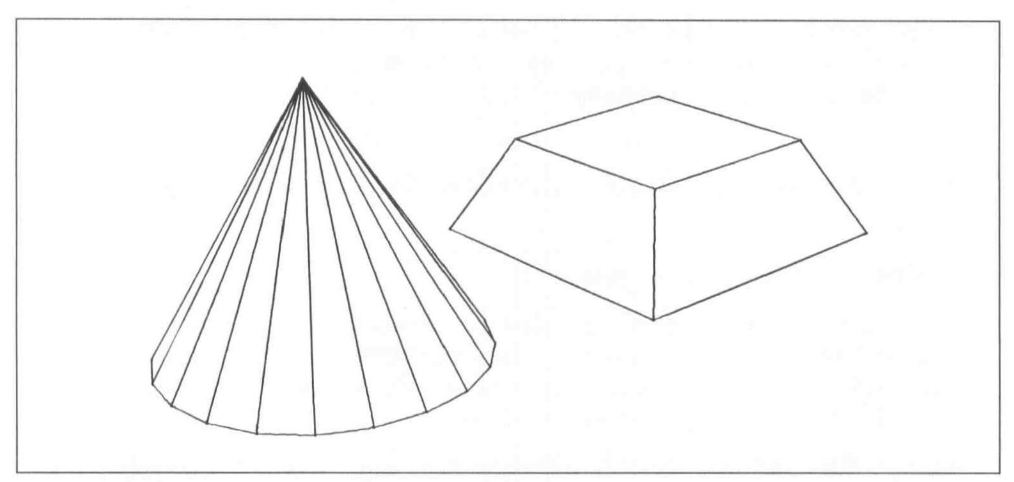


Abb.3.12: Pyramiden- und Kegelstumpf
a) Drehkegel
b) Pyramidenstumpf

Durch Manipulationen (also Translationen oder Rotationen) gelangt der Baustein an jede beliebige Stelle im Raum. Wenn wir mehr als zwei Meridianpunkte angeben, kann man auch komplizierte Objekte rasch beschreiben. Doch auch hier ist Vorsicht am Platz. Weil der 3-D-Sprinter konvexe Objekte so sehr bevorzugt, zwingt er den Ersteller eines Datenfiles dazu, daß der entstehende Rotationskörper zumindest einen konvexen Umriß hat. Der Datenblock

```
KONVEXE_DREHFLÄCHE gelb drehflaeche hohl
  meridian 2 0 -2, 4 0 0, 3 0 3, 1 0 5  rot_zahl 15
```

beschreibt die Drehfläche in Bild 3.13, während das Objekt in Bild 3.13 in drei Teile zerlegt werden muß:

```
KONVEXER_VASEN_UNTERTEIL gelb drehflaeche voll
  meridian 2 0 -2, 4 0 0, 3 0 3, 1 0 5  rot_zahl 15
VASEN_MITTELTEIL gelb drehflaeche voll
  meridian 1 0 5, 1 0 8  rot_zahl 15
VASEN_OBERTEIL gelb drehflaeche hohl
  meridian 1 0 8, 2.5 0 10 rot_zahl 15
```

Bild 3.13 illustriert, warum das so sein muß: Wenn nämlich der Innenwinkel des Meridianpolygons größer als 180 Grad wird, kann das Objekt nicht mehr konvex sein. In der nächsten Version des 3-D-Sprinters wird man sich darum nicht mehr kümmern müssen.

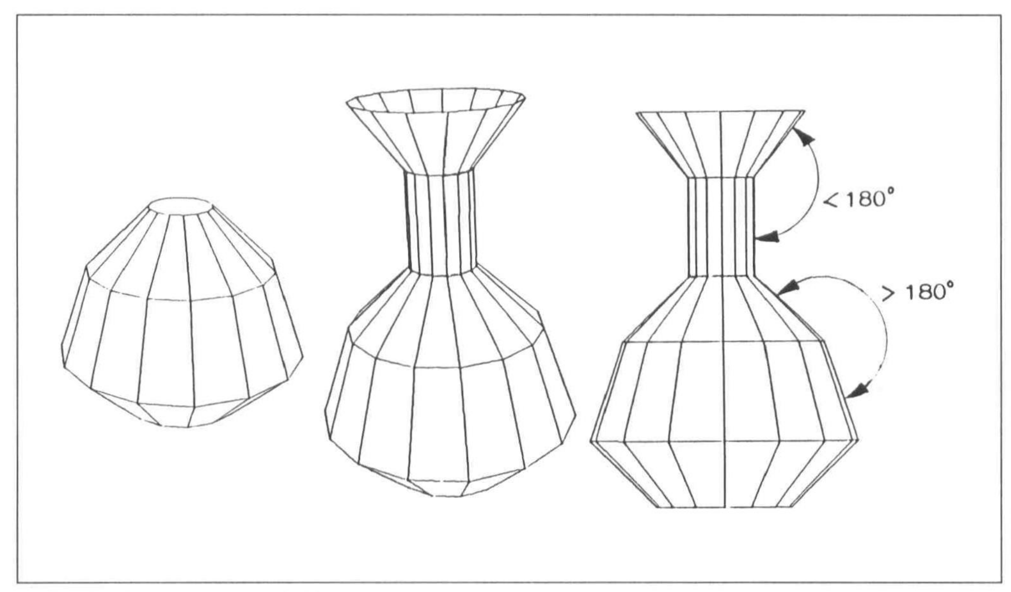


Bild 3.13: Konvexe und nichtkonvexe Drehflächen, die aus konvexen Teilen zusammengesetzt sind

Zum Listing ein Hinweis: Der Unterteil der Vase wurde als Vollkörper angenommen. Man kann ja sagen, die Vase sei bis zur Höhe z = 8 mit Wasser gefüllt. Der Vorteil ist, daß die Darstellung dann wesentlich rascher geht, insbesondere im Schlagschatten- und Hidden-Line-Modus. Fürs Auge ist praktisch kein Unterschied zu erkennen. Die Fläche in Bild 3.13 hat übrigens nur dann einen Boden, wenn Sie es auch hinschreiben:

```
EINE_KONVEXE_DREHFLÄCHE gelb drehflaeche hohl
  meridian 2 0 -2, 4 0 0, 3 0 3, 1 0 5  rot_zahl 15
  +boden
```

Analog bewirkt das Wörtchen »+deckel«, daß dem ganzen einen Deckel aufsetzt. Aber schreiben Sie jetzt bitte nicht »hohl +boden +deckel«, weil dann die Fläche nicht mehr von einem Vollkörper zu unterscheiden ist, was mit »voll« erstens viel einfacher ist, und zweitens sind Hohlobjekte »Rechenzeitkiller«, vor allem beim Schlagschatten- bzw. Hidden-Line-Modus!

Noch ein kleiner Zusatz: Wenn Sie nach dem Schlüsselwort »drehflaeche« das Wörtchen »winkel« anhängen, erwartet das Programm einen Öffnungswinkel der Drehfläche im Gradmaß (sonst werden automatisch 360 Grad genommen). Mit »winkel 180« kann man z. B. leicht Drehkörper herstellen, die in der Mitte durchgeschnitten sind (siehe dazu das Datenfile der Dorfkirche, wo der Kirchenanbau ein halber Drehkörper ist). Es wird allerdings kein Sektor aus dem Körper heraus-

geschnitten, sondern der Körper ebenflächig abgeschlossen (sonst wäre er nicht mehr konvex). Ein Beispiel dafür ist das Datenfile des Schraubstocks.

In Kap. 7.5.1 werden wir noch eine zweite Methode zur Erzeugung glatter Drehflächen kennenlernen. Diese Flächen werden dann als »mathematische Flächen« interpretiert und dementsprechend verarbeitet. Wenn immer es geht, ist aber die Methode der Zersplitterung in konvexe Teilobjekte vorzuziehen!

3.6.3 Der Typ »kugel«

Ein besonders einfacher Typus ist der Kugeltypus. Natürlich werden wir keine Kugel im mathematischen Sinn zeichnen, sondern eine Annäherung vornehmen. Dazu brauchen wir den Mittelpunkt, den Radius, und in Anlehnung an den vorigen Typus, die Anzahl der Meridianpunkte sowie die Rotationszahl:

```
MEINE_ERSTE_KUGEL grau kugel
  radius 4
  meridian_groesse 15   /* 15 Punkte werden rotiert */
  rot_zahl 20           /* und zwar 20 mal, also 300 Punkte */
  translation 5 3 2     /* Mittelpunkt */
```

Betrachten Sie einmal Bild 3.14 und das zugehörige Datenfile sowie die beiden zugehörigen Farbbilder im Anhang. Ist doch eindrucksvoll, nicht? BITTE: Übertreiben Sie nicht mit der Rotationszahl oder der Anzahl der Meridianpunkte. 500 Punkte auf einer Kugel sollten genügen, und die erhalten Sie mit »meridian_groesse 20« und »rot_zahl 25«.

```
STILLEBEN_MIT_KUGELN

augpunkt 35 25 40
licht 150 -20 100

5 bausteine

gruppe KUGELN_MIT_KEGEL {
   KEGEL gelb drehflaeche voll
      meridian 4 0 -2, 0 0 12 rot_zahl 20
   KUGEL1 grau kugel
      radius 5 meridian_groesse 10 rot_zahl 15
      translation -11.5 0 3.5
   KUGEL2 gruen wie KUGEL1 translation 17.32 10  0
   KUGEL3 gelb wie KUGEL1 translation 17.32 -10 0
}  translation x 3
TISCH grau quader voll 30 30 0.5  translation -15 -15 -2.5
```

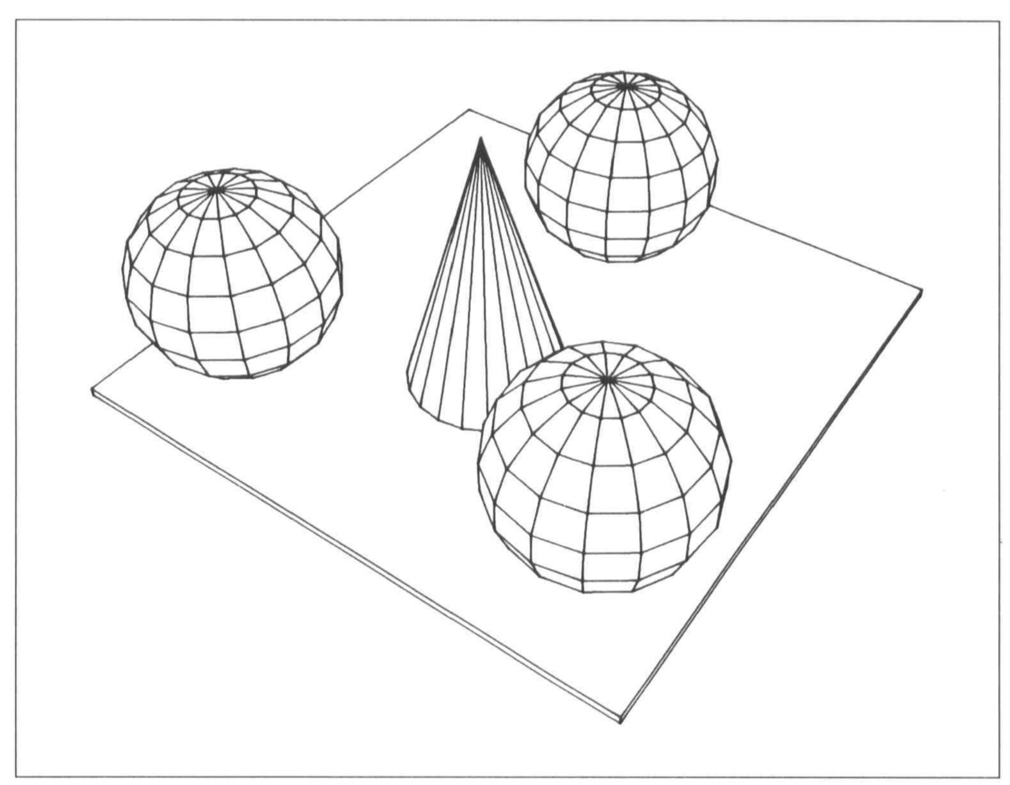


Bild 3.14: Kugeln mit Kegel

3.6.4 Der Typ »zylinder«

Wozu einen Typus »zylinder« definieren, wenn's doch mit dem Typus »drehflaeche« so einfach gegangen ist? Nun, dieser Typus erlaubt etwas mehr: Er kann »Zylinderhufe« erzeugen. Solche Teile kommen bei technischen Gegenständen oft vor. Bild 3.15 zeigt, was man unter den linken z-Werten bzw. den rechten z-Werten zu verstehen hat. Alles andere müßte klar sein. Betrachten Sie einmal das Datenfile für die »Kaffeemaschine« (der Griff!) oder jenes für das »Häuschen im Walde« (der Kamin!).

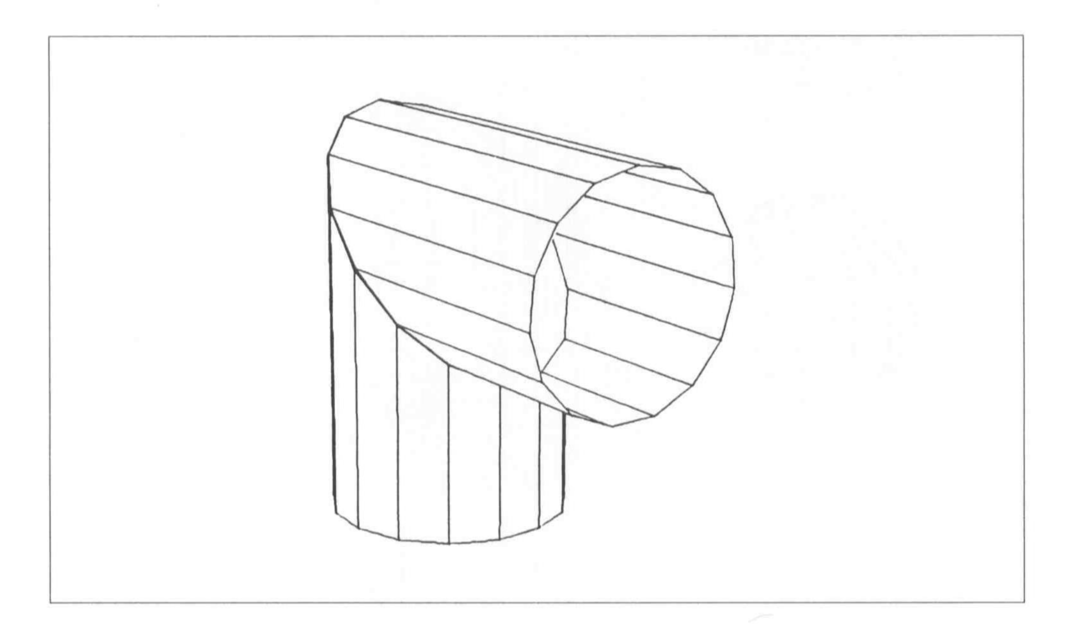


Bild 3.15: Rohrverbindung

Die Rohrverbindung in Bild 3.15 ist zum Aufwärmen allerdings leichter verständlich. Das Datenfile dazu lautet:

```
ROHR_1 grau zylinder hohl
   radius 3
   z_links  -7  3 /* die linken Höhenwerte  */
   z_rechts -7 -3 /* die rechten Höhenwerte */
   rot_zahl 15
ROHR_2 grau wie ROHR_1
   rotation x 90 rotation z 180
```

Wenn doch alles so leicht ginge ...

3.6.5 Der Typ »translation«

Ein nächster überaus wichtiger Typus umfaßt jene Bausteine, die entstehen, wenn man eine Basisfigur im Raum parallel verschiebt. Dazu gehören als einfachste Spezialfälle natürlich auch der Würfel (Parallelverschieben eines Quadrats oder regelmäßige Prismen (Parallelverschieben eines regelmäßigen Vielecks). Nur können diese Bausteine einfacher mit dem Typus »quader« bzw. »drehflaeche« beschrieben werden.

Bild 3.16 zeigt einen Gebäudekomplex, der durch Entlangziehen des Grundrisses in z-Richtung entsteht. Bei jedem einzelnen Baustein muß das Basispolygon allerdings unbedingt wieder konvex sein, damit die Sichtbarkeit stimmt. Die nächste

Version des 3-D-Sprinters wird in der Lage sein, nichtkonvexe Basispolygone in konvexe zu zersplittern. Die zugehörigen Datenzeilen sehen wie folgt aus:

```
ZENTRUM gelb drehflaeche voll
   meridian 1.5 0 2, 1.5 0 8 rot_zahl 6
FLUEGEL1 grau translation voll
   basis 0.75 1.30 2, 0.35 4 2, -0.35 4 2, -0.75 1.30 2
schub_vektor z 6
FLUEGEL2 grau wie FLUEGEL1 rotation z 120
FLUEGEL3 grau wie FLUEGEL1 rotation z 240
STUETZE1 gelb  box voll 3 0.5 9  translation -1.5 4 0
STUETZE2 gelb  wie STUETZE1 rotation z 120
STUETZE3 gelb  wie STUETZE1 rotation z 240
```

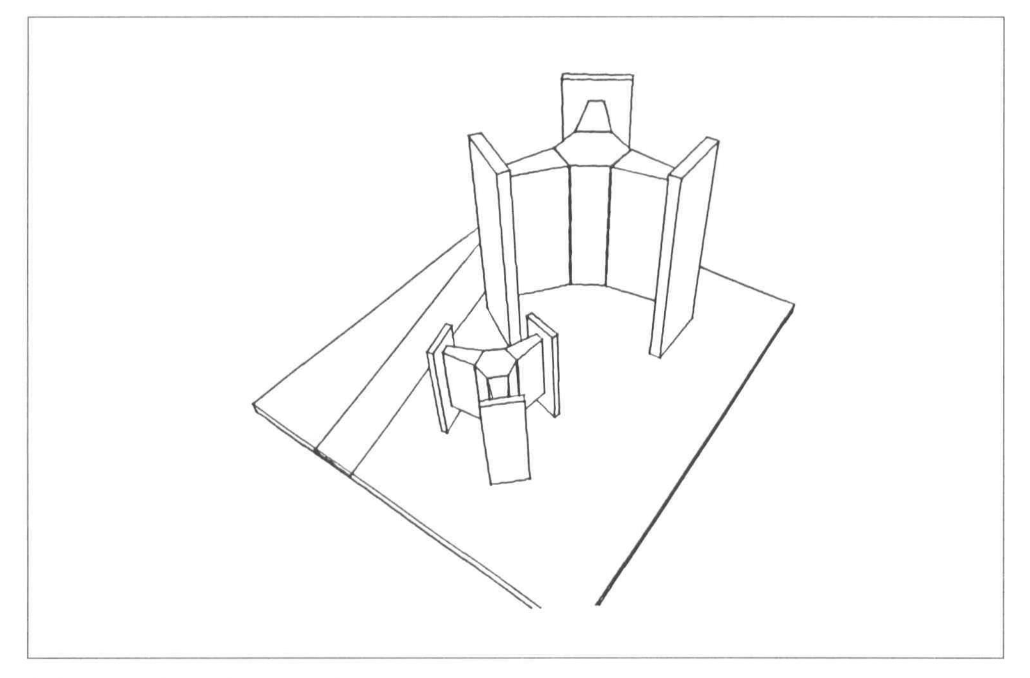


Bild 3.16: Gebäudekomplex, bestehend aus Translationskörpern

Die Schlüsselwörter sind offensichtlich »basis« und »schub_vektor«. Diesen Wörtern folgt eine Auflistung der Koordinaten des Basispolygons bzw. eine Angabe des Schubvektors, die wie beim Schlüsselwort »translation« erfolgen kann.

Analog zu den besprochenen Typen können die Bausteine auch hohl sein, wobei dann die Zusatzoptionen »+boden« bzw. »+deckel« möglich sind, was praktisch für die Erzeugung von »Schachteln« ist (Bild 3.17).

```
SCHACHTEL translation hohl
   basis -3 0 0, -2 -2 0, 2 -2 0, 3 0 0, 2 2 0, -2 2 0
   schub_ vektor x 2
   +boden
```

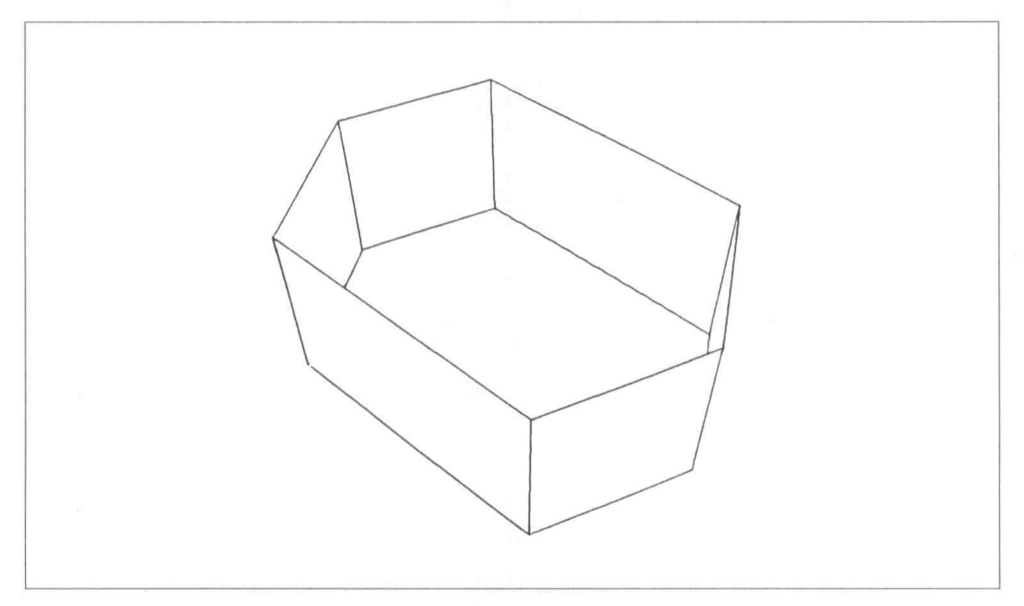


Bild 3.17: Offener Translationskörper

3.6.6 Der Typ »allgemein«

Was aber, wenn sich der Baustein, den wir erzeugen wollen, in kein Schema einordnen läßt? Tja, dann müssen wir für den Computer das erledigen, was er üblicherweise ganz von selbst macht:

1) Anlegen einer Punkteliste, in der die Koordinaten aller Punkte angeführt sind,
2) Anlegen einer Kantenliste, in der steht, welcher Punkt mit welchem zu verbinden ist,
3) Anlegen einer Flächenliste, in der steht, welche Punkte zu Facetten zusammenzufassen sind.

Leider geht's nicht anders. Gott sei Dank lassen sich die meisten technischen Gegenstände irgendwo einordnen. Als Musterbeispiel soll eine unregelmäßige dreiseitige Pyramide angeführt werden:

```
UNREGELMAESSIGE_PYRAMIDE allgemein voll
punkte 3 0 0 , -1 2 0 , -1 -2 0 , 0 0 5
kanten 0 1 , 1 2 , 2 0 , 0 3 , 1 3 , 2 3
flaechen 0 1 2 , 0 1 3 , 1 2 3 , 2 0 3
```

Die Erklärung der Datenzeilen ist unschwer aus Bild 3.18 abzulesen.

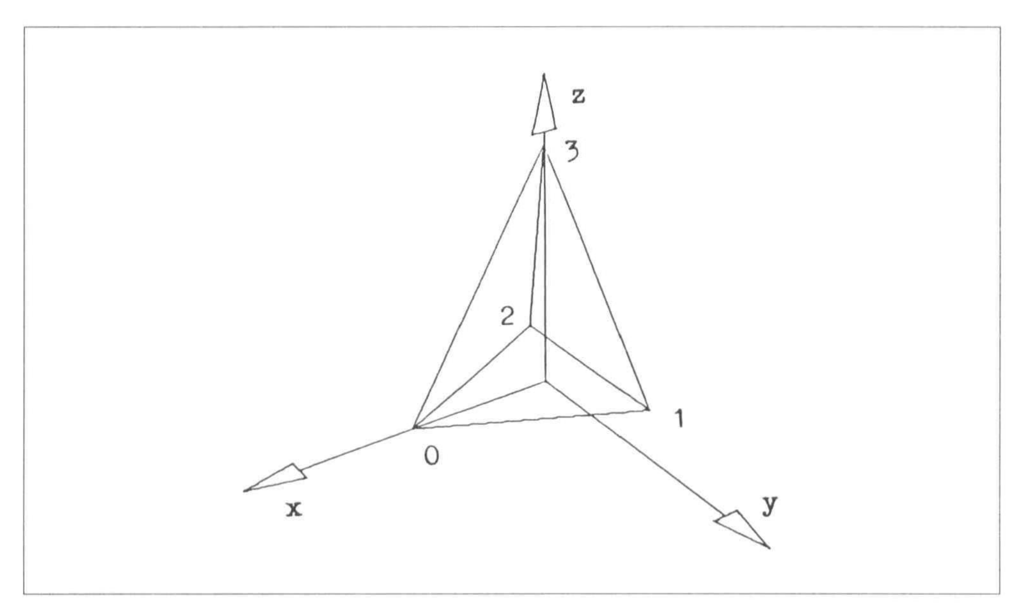


Bild 3.18: Ein allgemeiner Körper

Noch ein wichtiger Hinweis: Wenn Sie wirklich das dringende Bedürfnis verspüren, komplizierte allgemeine Objekte darzustellen, dann sollten Sie unbedingt beachten, daß jeder einzelne Baustein konvex ist. Nichtkonvexe Objekte müssen folglich in mehrere konvexe aufgesplittert werden. Für Ihre Mühe belohnt Sie allerdings der 3-D-Sprinter mit seinen kurzen Rechenzeiten!

3.6.7 Der Typ »polygon«

Ein Spezialfall eines allgemeinen Körpers ist ein Polygon, also ein Körper, der nur mehr aus einer Fläche besteht. Wir könnten demnach z. B. ein Dreieck beschreiben, indem wir etwa folgende Zeilen hinschreiben:

```
IRGENDEIN_DREIECK gruen allgemein hohl
   punkte 1 2 3, 2 3 5, -1 2 5
   kanten 0 1, 1 2, 2 0
   flaechen 0 1 2
```

Weil die Eigenschaft »hohl« und die Kantenliste bzw. Flächenliste trivial ist, kann man das ganze mit einer Zeile bewerkstelligen:

```
IRGENDEIN_DREIECK gruen
    polygon normal 1 3 -1 bezugspunkt 0 0 2
            punkte 1 2, 2 3, -1 2
```

Nach dem Wörtchen normal werden die Komponenten des Normalenvektors des Polygons erwartet. Wenn Sie das Schlüsselwort »bezugspunkt« (samt dessen Koordinaten) weglassen, wird automatisch der Koordinatenursprung als Bezugspunkt gewählt. Die Punkte werden nun in einem zweidimensionalen Koordinatensystem angegeben, dessen Ursprung der Bezugspunkt ist und dessen Achsen zusammen mit der Flächennormalen ein »rechtwinkeliges Dreibein« bilden (siehe dazu auch Kapitel 7.3). Insbesondere lassen sich Quadrate und Rechtecke wie folgt definieren:

```
MEIN_QUADRAT gruen
   quadrat normal x bezugspunkt -2 -2 -2 seite 4
MEIN_RECHTECK gelb
   rechteck normal 3 2 1 /* bezugspunkt 0 0 0 */seiten 2 6
```

Die Worte »quadrat« und »rechteck« sind Schlüsselwörter. Nach dem Wort »normal« folgt wie oben die Angabe der Flächennormalen und nach dem Wort »bezugspunkt« ein Eckpunkt des Quadrats (Rechtecks). Anschließend werden die Seitenlängen eingegeben.

3.7 Zusammenfassen der Bausteine zu Gruppen

Der 3-D-Sprinter hat einen schnellen Sortieralgorithmus eingebaut, der die einzelnen Bausteine so sortiert, daß beim Zeichnen in der richtigen Reihenfolge durch Übermalen einzelner Flächenteile ein richtiges Bild entsteht. Entscheidend für das richtige Sortieren ist, daß sich die Bausteine nicht überlappen.

Sollte es Ihnen passieren, daß ein Bild nicht korrekt gezeichnet wurde, dann editieren Sie bitte das »Nachrichtenfile« und sehen Sie sich an, was da drinnen steht. Es werden höchstwahrscheinlich zwei Meldungen folgender Art enthalten sein: Probleme beim Sortieren. Dies liegt wahrscheinlich daran, daß sich einzelne Bausteine überschneiden!

***Achtung**: Bausteine können nicht sortiert werden! Richtige Sichtbarkeit ist nicht mehr garantiert!*

Dann müssen Sie versuchen, dies zu verhindern, indem Sie zusätzliche Manipulationen (Translationen oder Rotationen) auf die kritischen Bausteine anwenden.

Der seltene Fall wird sein, daß ein richtiges Sortieren tatsächlich unmöglich ist (dann würde die erste Meldung fehlen!). Bild 3.19 zeigt einen solchen hartnäckigen Fall. Auch diese Fälle lassen sich kurieren, indem große Objekte einfach zersplittert werden (Bild 3.19). Der Hidden-Line-Modus (h-Taste) müßte sich allerdings von solchen Problemen unbeeindruckt zeigen (das bedeutet, daß beide Bilder vom Plotter korrekt ausgegeben werden, beim shade-Modus aber Fehler auftreten).

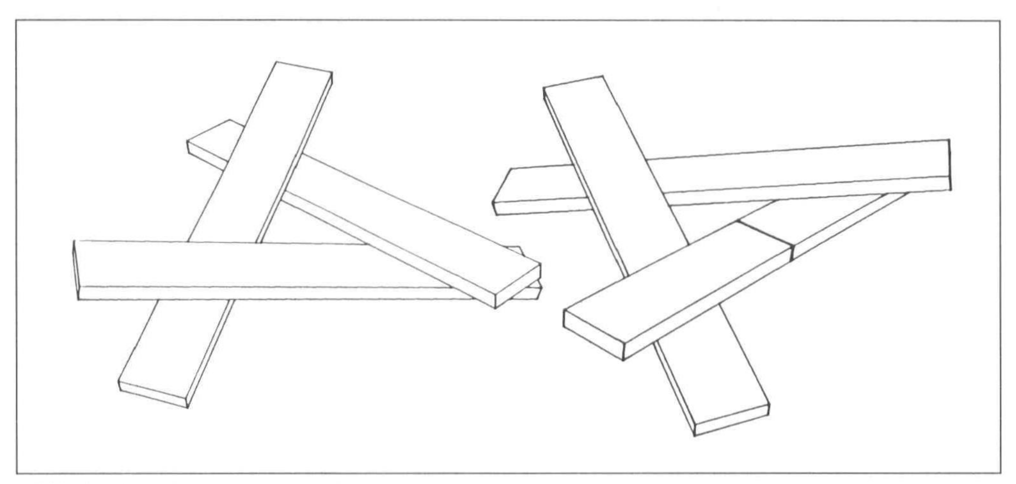


Bild 3.19: Der (eher seltene) Alptraum beim Erstellen von Datenfiles: Ein Sortieren ist in a) nicht möglich. Durch einen Trick wie in b) geht's dann doch wieder

Wenn Sie nun eine Bausteingruppierung aus sagen wir 40 Bausteinen zusammengestellt haben und keine Überschneidungen auftreten, sollte eigentlich alles klargehen. Sie freuen sich darüber, daß der 3-D-Sprinter immer noch verhältnismäßig schnell zeichnet. Dabei könnte es noch ein wenig schneller gehen, wenn Sie es geschickt angehen:

Der 3-D-Sprinter hat nämlich die Möglichkeit eingebaut, kleinere Gruppen von Bausteinen zusammenzufassen. Dies geschieht ganz einfach dadurch, daß Sie wie im folgenden Beispiel

```
gruppe LINKER_GEBAUEDETEIL {
  GARAGE ...
  VORBAU ...
  BALKON ...
}
```

einfach das Schlüsselwort »gruppe« plus einen Namen für diese Gruppe ins Datenfile schreiben und anschließend die Objekte, die Sie zusammenhalten wollen, in geschwungene Klammern geben. Dies bringt, abgesehen von einer übersichtlichen Strukturierung des Datenfiles, zusätzliche Vorteile:

1) Sie können nach dem Schließen der geschwungenen Klammer Manipulationen (Rotationen, Translationen, Streckungen) auf die gesamte Gruppe anwenden.
2) Wenn Sie es geschafft haben, die Bausteine so in Gruppen einzuteilen, daß sich die Koordinatenquader, die man den Gruppen zuschreiben kann, nicht schneiden, wird der Sortieralgorithmus noch schneller: Der 3-D-Sprinter sortiert dann zuerst die Bausteine innerhalb der Gruppen und dann die Gruppen

untereinander. Rechenbeispiel bei unseren 40 Bausteinen mit sagen wir fünf Gruppen zu je acht Bausteinen: Statt etwa 800 Vergleichen braucht das Programm nur mehr etwa ein Drittel der Vergleiche durchzuführen. Bei noch mehr Bausteinen wird das Verhältnis noch extremer!

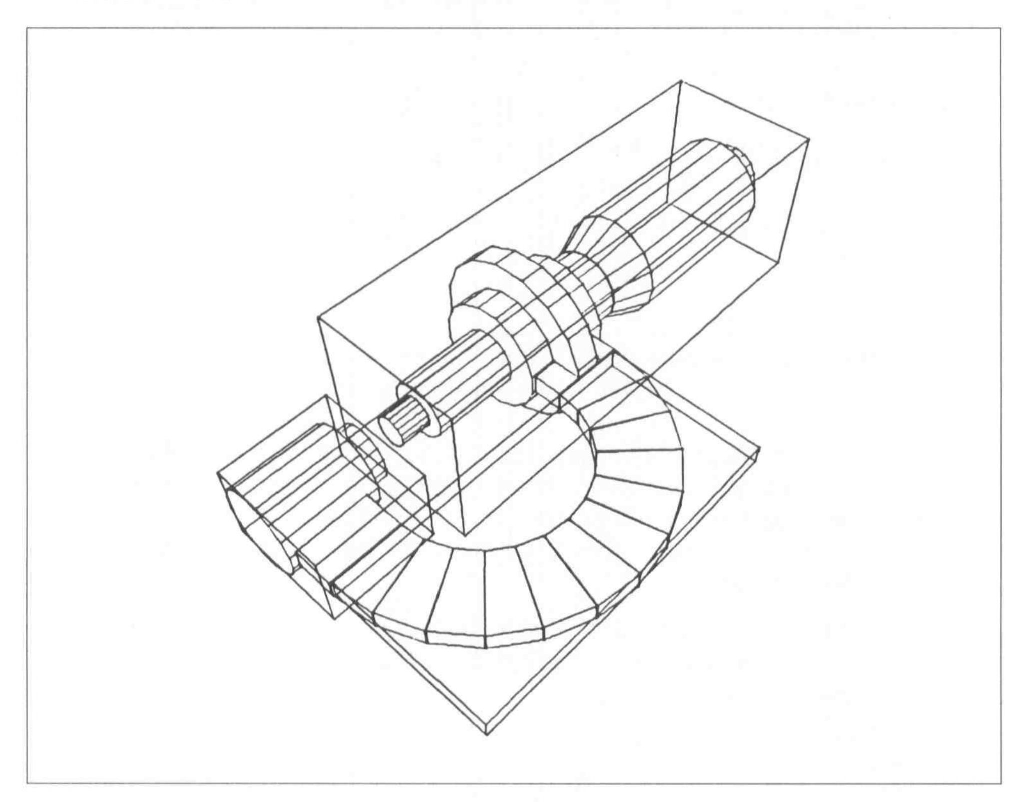


Bild 3.20: Einteilung in Gruppen zur besseren Strukturierung und zur Beschleunigung des Sortierverfahrens

3) In der nächsten Version des 3-D-Sprinters wird es möglich sein, Bausteine während des Programmablaufs mittels Animations-Skripten zu manipulieren. Meist möchte man aber ganze Bausteingruppen bewegen oder in der Größe verändern. Das Schreiben der Animations-Skripten wird bei Verwendung von Gruppen wesentlich bequemer.

3.8 Der 3-D-Sprinter für wissenschaftliche Anwendungen

Eine weitere Stärke des 3-D-Sprinters ist seine Fähigkeit, gewisse mathematische Flächen schnell zu zeichnen und Kurven auf solchen Flächen selbständig aufzuspüren: In der Kartographie, aber auch in gewissen Anwendungsbereichen der Physik und Elektrotechnik benötigt man von Funktionsgraphen nicht nur Parameterlinien, sondern vor allem Höhenschichtenlinien (Linien gleichen Potentials, Linien gleicher Spannungsintensität usw.). Durch die Datazeile

```
schichtenlinien 20
```

nach dem Datenteil für den Funktionsgraphen (oder jede andere mathematische Fläche) wird das Programm veranlaßt, den Höhenschichtenbereich, der durch die

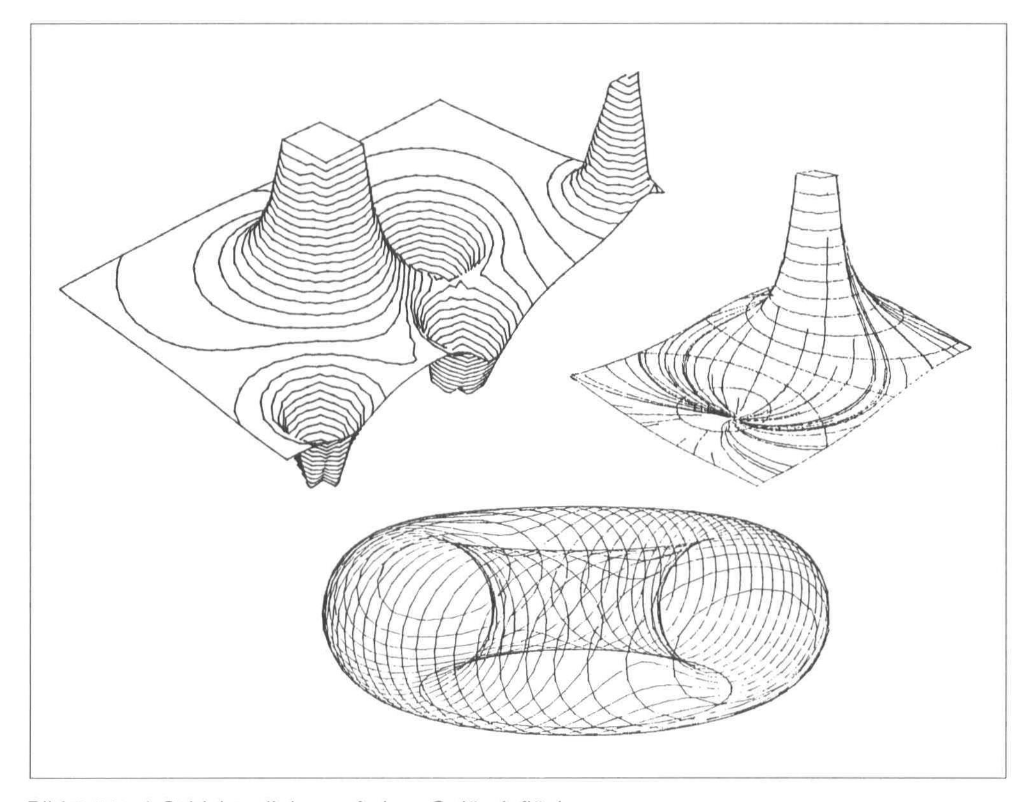


Bild 3.21: a) Schichtenlinien auf einer Geländefläche
b) Potentiallinien auf einer Potentialfläche
c) Parallelschnitte eines Torus

Fläche vorgegeben ist, in 19 gleiche Schichten zu zerlegen und für die insgesamt 20 Schichtenhöhen die Schnittlinien mit der Fläche einzuzeichnen. Die Rechenzeit dafür bewegt sich im Bereich weniger Sekunden, wobei die Kurven abgespeichert werden, so daß die Flächen anschließend ohne Zeitverlust interaktiv manipuliert werden können. Durch einen kleinen Trick gelangt man auch zu Schnittlinien der Fläche mit Ebenen, die nicht mehr horizontal sind: Man braucht nur die Fläche selbst einer Rotation zu unterwerfen! Um das Bild wieder »aufzurichten«, genügt die Angabe eines Verdrehungswinkels.

Genaueres über die mathematischen Flächen lesen Sie bitte in Kapitel 7.5.2.

3.9 Zukunftsmusik für eine erweiterte Version

Wie Sie bereits gesehen haben, genügt dem 3-D-Sprinter ein Schlagwort wie etwa »quader«, und schon zaubert er – zusammen mit ein paar zusätzlichen Längenangaben – eine Punkteliste, Kantenliste und Flächenliste hervor. Je mehr solcher Schlüsselwörter das Programm erkennt, desto einfacher wird es, die gewünschten Objekte mit wenigen Datenzeilen zu erfassen. Ein Beispiel: Auf das Schlagwort »minipalme« wäre es möglich, eine standardisierte Topfpflanze (etwa eine Yuka-Palme) zu definieren. Mit dem angebbaren Größenfaktor und den möglichen Translationen kann dann überallhin eine Topfpflanze postiert werden.

Wie Sie beim Herumspielen mit den Datenfiles vielleicht schon bemerkt haben, zeichnet der 3-D-Sprinter bei komplizierten mathematischen Flächen keine Schlagschatten mehr ein. Das in Kapitel 3.5 beschriebene Darstellungsverfahren ist sehr wohl dazu in der Lage, doch irgendwann kommt der Punkt, wo man mit einer Erstversion auf den Markt kommen muß.

Die schnellen Algorithmen lassen sich außerdem auch auf andere Feinheiten der grafischen Darstellung ausdehnen, etwa auf Spiegelungen und durchsichtige Objekte. Doch bis es soweit ist, werden noch einige Nächte voll mit Programmierarbeit dran glauben müssen. Immerhin, die Spiegelung an einer einzelnen Fläche (ohne Einzeichnen der Schlagschatten) wurde bereits während der Fertigstellung dieses Buches realisiert.



Die Bedienung des Programms

4.1 Programmstart

4.1.1 Start von der Workbench

Wenn Sie den Amiga mit der 3-D-Sprinter-Programmdiskette gestartet haben, genügt ein Doppel-Klick auf das Piktogramm »3-D-Sprinter«.

Bild 4.1: Der Inhalt der 3-D-Sprinter-Schublade

Wenn nicht, müssen Sie zuerst das Piktogramm »Assign_3-D-Sprinter« zweimal schnell hintereinander anklicken. Wenn Sie sich diese zusätzliche Arbeit ersparen wollen, lesen Sie im Anhang nach, wie Sie die Diskette (oder Festplatte), mit der Sie den Amiga starten, modifizieren müssen.

4.1.2 Start vom CLI oder der Shell

Zuerst müssen Sie mit dem Befehl »cd« in das richtige Verzeichnis wechseln. Danach bitte den Befehl

```
Assign 3-D-Sprinter: ""
```

eingeben. Dies ist notwendig, da das Programm über das Pseudo-Laufwerk 3-D-Sprinter auf wichtige Daten zugreift. Nun gibt es zwei Möglichkeiten, das Programm zu starten: Wenn Sie mit der grafischen Benutzeroberfläche arbeiten wollen, geben Sie »3-D-Sprinter« ein.

Ist Ihnen das Hantieren mit den Menüs zu umständlich oder zu zeitaufwendig, können Sie den Animator auch direkt starten. Die Befehls-Syntax ist:

```
try Datenfile [-a Datei] [-v Datei | +v Datei] [-i Datei]
              [-c Datei] [-s]
```

Der Name des Datenfiles muß immer an erster Stelle nach dem Programmnamen stehen. Findet »try« das Datenfile nicht, wird noch im Verzeichnis »3-D-Sprinter/datas« nachgesehen. Dies können Sie ausnützen, wenn Sie sich Tipp-Arbeit ersparen wollen. Die optionalen Angaben (durch die eckigen Klammern gekennzeichnet) können in beliebiger Reihenfolge stehen und haben folgende Bedeutung:

`-a Datei`

Um die Animation zu steuern, wird das durch »Datei« bezeichnete Animations-Skript verwendet.

`-v Datei`

Wenn Sie nach dem Start des Programms die [v]-Taste betätigen, werden alle folgenden Bilder als Video unter dem Namen »Datei« abgespeichert.

`+v Datei`

Wie oben, es wird jedoch ab dem ersten Bild abgespeichert. Diese Option ist besonders in Verbindung mit Animations-Skripten wichtig, denn nur so kann ab dem ersten Bild abgespeichert werden.

`-c Datei`

3-D-Sprinter nimmt anstelle der Standard-Farbpalette die durch »Datei« angegebene.

`-i Datei`

Um ein Bild im IFF-Standard abspeichern zu können, muß natürlich sein Name (»Datei«) bekannt sein.

`-s`

Diese Option (»silent« bzw. still) veranlaßt try, seine Ausgaben (Fehlermeldungen) in die Datei »t:message« zu schreiben. Dies ist sicher praktisch, wenn Sie die Ausgabe mehrmals sehen wollen. Hier einige Beispiele: Mit

```
try Kaffee.data -v video/Kaffee_1.video
```

wird die Kaffeekanne dargestellt und kann anschließend interaktiv manipuliert und jederzeit durch Drücken von [v] als Video abgespeichert werden.

```
try Kirche.data -a anim_scripts/Sonne.anim -i df1:kirche
```

bewegt die Dorfkirche wie in dem Animations-Skript »Sonne.anim« im Verzeichnis »anim_scripts« angegeben. Weiterhin kann ein beliebiges Bild durch Drücken von [i] unter dem Namen Kirche auf die Diskette im Disketten-Laufwerk »DF1:« abgespeichert werden.

4.1.3 Allgemeines

Die grafische Benutzeroberfläche können Sie praktisch völlig mit der Maus bedienen (manche Eingaben, z. B.: Filenamen, Editieren des Datenfiles, ... gehen natürlich nur über die Tastatur). Fast alle Funktionen können Sie aber auch über die Tastatur aufrufen (dies geht, wenn Sie eine gewisse Übung haben, auch schneller). Nach dem Starten des Programms sehen Sie einige Copyright-Meldungen und die Erklärung des Programmnamens eingeblendet.

Dieser Vorgang wird abgebrochen, sobald Sie den Maus-Pfeil in den Bereich der Menü-Leiste schieben (also mit der Arbeit begonnen wird). Nun beginnt sich der Schriftzug des Maus-Pfeiles (3-D-Sprinter) zu bewegen, zum Zeichen, daß das Programm darauf wartet, daß Sie einen Menüpunkt anwählen oder ein Symbol anklicken. Wenn eine Funktion angewählt wird, gibt es drei Möglichkeiten:

1) Die Schrift wird zu »BUSY«: Das Programm ist gerade beschäftigt und sollte nicht gestört werden (der Guru könnte sonst grüßen; dies ist zwar übertrieben, dennoch sollten Sie den Amiga während Diskettenoperationen nicht stören).
2) Der Mauszeiger rotiert weiter: Eine Eingabe in dem erschienenen Kommunikationsfenster wird erwartet.
3) Der Mauszeiger wird wieder zum Maus-Pfeil der Workbench: Sie haben eine Funktion des Programms angewählt, die ein Systemprogramm (z. B.: Preferences (den Voreinsteller), einen Texteditor, ...) aufruft.

Auch das Starten des Animators und des »Replay«-Programms normalisiert den Maus-Pfeil. Im Grundprinzip werden alle Funktionen über Menüs (rechte Maus-Taste) aufgerufen bzw. aktiviert.

Nun folgt eine Beschreibung der einzelnen Menüs und Menüpunkte. Der Umgang mit den Kommunikationsfenstern wird dabei gleich miterklärt. Die Vorgehensweise ist praktisch immer gleich: Das Kommunikationsfenster über den entsprechenden Menüpunkt aufrufen (aktivieren), die Werte verändern und das Kommunikationsfenster wieder schließen.

4.2 Das »Datei«-Menü

In diesem Menü sind alle disketten- und programminternen Funktionen untergebracht.

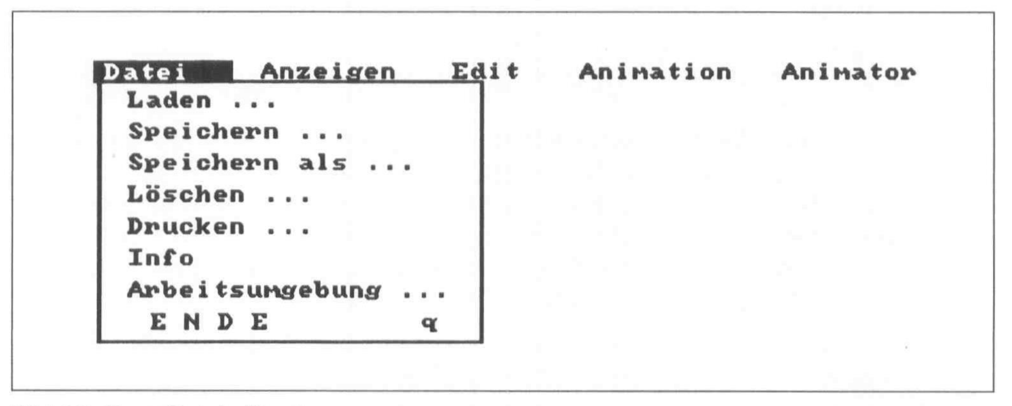


Bild 4.2: Das »Datei«-Menü

4.2.1 Datei->Laden ...

Über diesen Menüpunkt können Sie alle Daten laden bzw. Dateien öffnen. Über einen der vier Untermenüpunkte wird festgelegt, welche Daten geladen werden sollen.

Datenfile: Bewirkt das Laden des Datenfiles, in dem das Aussehen der Objekte bestimmt wird. Hierzu öffnet sich das Dateiauswahl-Kommunikationsfenster, das im nächsten Kapitel beschrieben wird.

Animations-Skript: Über diesen Menüpunkt können Sie ein Animations-Skript laden, welches die Bewegung der Objekte festlegt.

Video: Um ein bereits erstelltes Video wieder abspielen oder überspielen zu können.

Farb-Palette: Wenn Sie einmal andere als die Standard-Farben für eine Animation benutzen wollen, können Sie diese über diesen Menüpunkt laden.

4.2.2 Das Dateiauswahl-Kommunikationsfenster

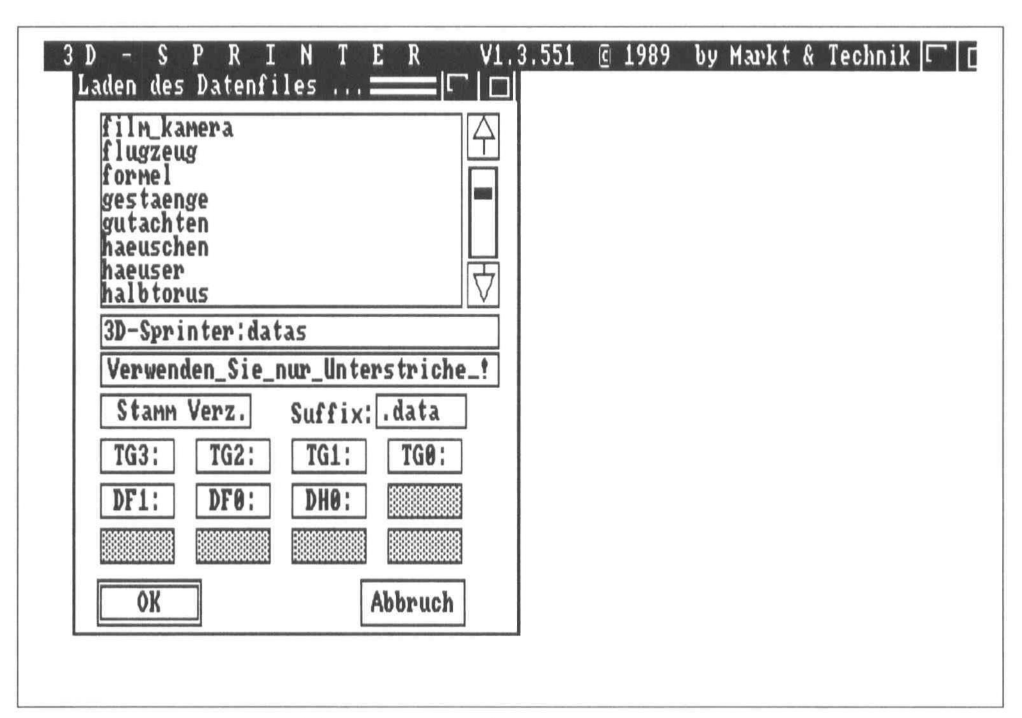


Bild 4.3: Das Dateiauswahl-Kommunikationsfenster

In der oberen Hälfte wird der Inhalt des aktuellen Verzeichnisses (Directory) in alphabetischer Reihenfolge angezeigt (zuerst die Unterverzeichnisse, dann die Dateien). Da in den vorhandenen acht Zeilen vermutlich nur selten alle Dateien und Unterverzeichnisse dargestellt werden können, befinden sich rechts neben den Dateinamen drei Symbole, mit denen die Liste der Einträge bewegt werden kann.

Die beiden Pfeile bewirken, wenn sie kurz angeklickt werden, das Verschieben der Liste in die Richtung, in die ihre Pfeilspitze zeigt. Hält man hingegen die Maus-Taste über dem Symbol gedrückt, wird die Liste langsam in die gewünschte Richtung bewegt. Der Schieberegler zwischen den beiden Pfeil-Symbolen kann zum freien Verschieben der Liste in beide Richtungen benutzt werden. Durch ihn ist es möglich, sehr schnell zum Ende der Liste zu gelangen.

Ein Dateiname wird ausgewählt oder in ein Unterverzeichnis verzweigt, indem der entsprechende Name angeklickt, oder die Nummer der Zeile (1–8) über die Tastatur eingegeben wird. Unter der Anzeige der Dateinamen befinden sich drei Kästchen. Im obersten werden das aktuelle Laufwerk und der Pfad angezeigt. Das

Kästchen darunter enthält den Dateinamen, und rechts unterhalb wird auch noch die Endung (Suffix) des Dateinamens angezeigt.

Wie Ihnen vielleicht schon aufgefallen ist, wird bei jedem Dateityp (Datenfile, Animations-Skript, ...) automatisch in das richtige Verzeichnis verzweigt. Um nicht unnötig viele Dateinamen anzuzeigen, wurde für jeden Dateityp eine spezielle Endung festgelegt. Hier eine kleine Übersicht:

Dateityp	**Verzeichnis**	**Endung**
Datenfiles	3-D-Sprinter:datas	.data
Animations-Skripten	3-D-Sprinter:anim_scripts	.anim
Videos	3-D-Sprinter:video	.video
IFF-Bilder	3-D-Sprinter:iff_pics	.iff
Farb-Paletten	3-D-Sprinter:color_maps	.color

Den Dateinamen und die Endung können Sie verändern bzw. neu eingeben, indem Sie mit der Maus in das entsprechende Rechteck klicken. Danach können Sie den Inhalt mit der Tastatur verändern. Mit [Return] beenden Sie die Eingabe.

Eine Aktivierung ist auch über die Tastatur möglich. Mit der Leer-Taste ([Space]) wird das Eingabefeld des Dateinamens aktiviert, [s] für »Suffix« ermöglicht die Veränderung der Endung, und mit [p] aktivieren Sie das Feld zur Eingabe des Pfades. Um von einem Unterverzeichnis in das darüberliegende Verzeichnis aufzusteigen, müssen Sie das »Stamm Verz.«-Symbol anklicken und die [/]-Taste drücken.

Um auch auf andere Laufwerke zugreifen zu können, werden im unteren Drittel des Kommunikationsfensters alle verfügbaren Laufwerke angezeigt. Wenn Sie eines dieser Symbole anklicken, wird das entsprechende Laufwerk zum aktuellen. Ein Klick auf das »O.K.«-Symbol (oder [Return]) führt die Funktion (Laden, Speichern, Löschen) aus, das »Abbruch«-Symbol (oder [Esc]) führt zum Abbruch der Funktion.

Wird das Dateiauswahl-Kommunikationsfenster erneut aufgerufen, befinden Sie sich wieder in dem Verzeichnis, das beim Verlassen aktiv war. Wenn Sie versehentlich ein falsches Verzeichnis oder Laufwerk angeklickt haben, können Sie bereits während des Einlesens des Inhalts den Fehler korrigieren. Auch ein Abbruch ist während des Einlesens möglich. Hier noch einmal eine Darstellung des Dateiauswahl-Kommunikationsfensters. Diesmal sind aber die Tasten eingezeichnet, die den Symbolen entsprechen (vgl. Bild 4.4).

4.2.3 Datei->Speichern ...

Um nicht jedesmal das Dateiauswahl-Kommunikationsfenster aufrufen zu müssen, können alle Daten auch unter dem Namen, unter dem sie geladen bzw. zuletzt abgespeichert wurden, über diesen Menüpunkt gespeichert werden. Wenn Sie

noch keinen Namen eingegeben haben, werden Sie durch ein Kommunikationsfenster darüber informiert. Bestätigen Sie den Erhalt der Mitteilung durch Anklicken eines der beiden »O.K.«-Symbole (vgl. Bild 4.5).

Bild 4.4: Tastaturbelegung im Dateiauswahl-Kommunikationsfenster



Bild 4.5: Fehlermeldung bei Disketten-Operationen

Hier noch einmal die Beschreibung der Untermenüpunkte:

Datenfile: Das aktuelle Datenfile wird unter dem in »Datei->Speichern als ...-> Datenfile« eingegebenen Namen oder unter dem Namen, unter dem es geladen wurde, abgespeichert.

Animations-Skript: Das aktuelle Animations-Skript wird unter dem im »Datei-> Speichern als ...->Animations-Skript« eingegebenen Namen oder unter dem Namen unter dem es geladen wurde, abgespeichert.

Farb-Palette: Die gerade in Verwendung stehende Farb-Palette wird unter dem in »Datei->Speichern als ...->Farb-Palette« eingegebenen Namen oder unter dem Namen, unter dem sie geladen wurde, abgespeichert.

4.2.4 Datei->Speichern als ...

Über diesen Menüpunkt können Sie festlegen, unter welchen Namen die Daten abgespeichert werden. Die Untermenüpunkte legen fest, welche Daten abgespeichert werden. Dazu erscheint das in Abschnitt 4.2.2 beschriebene Dateiauswahl-Kommunikationsfenster. Hier die zur Wahl stehenden Untermenüpunkte:

Datenfile: In diesem Kommunikationsfenster kann ein Filename, unter dem das Datenfile abgespeichert werden soll, eingegeben werden.

Animations-Skript: In diesem Kommunikationsfenster kann ein Filename, unter dem das Animations-Skript abgespeichert wird, eingegeben werden. Die Bedienung entspricht genau dem Kommunikationsfenster für die Datenfiles.

Video: Da ein Videofile sehr schnell sehr lang werden kann, wird es nicht zuerst temporär im Speicher erzeugt und danach auf Diskette gespeichert, sondern sofort, wenn es erzeugt wird, auf die Diskette (oder Festplatte) geschrieben. Um dies zu ermöglichen, müssen Sie über diesen Menüpunkt zuerst den Namen festlegen, unter dem gespeichert werden soll.

Farb-Palette: Wenn Sie eine eigene Farb-Palette erstellt oder eine vorhandene verändert haben, können Sie sie über diesen Menüpunkt unter einem neuen Namen abspeichern.

IFF-Bild: Wenn Sie ein Bild, das der Animator erzeugt hat, als IFF-Bild abspeichern wollen, müssen Sie aus demselben Grund wie bei den Videos vor dem Start des Animators den Namen angeben, unter dem das Bild gespeichert werden soll. Genau dazu dient dieser Menüpunkt.

4.2.5 Datei->Löschen ...

Wenn Sie Dateien, die Sie nicht mehr benötigen, löschen wollen, können Sie diesen Menüpunkt benützen. Über den entsprechenden Untermenüpunkt legen Sie fest, ob Sie

Datenfile – Datenfiles
Animantions-Skript – Animations-Skripten
Video – Videos
Farb-Palette – Farb-Paletten
oder
IFF-Bild – IFF-Bilder

löschen wollen. Es öffnet sich das Dateiauswahl-Kommunikationsfenster, und Sie können das File auswählen, das gelöscht werden soll. Überlegen Sie zweimal, bevor Sie das »O.K.«-Symbol anwählen, es gibt kein Zurück mehr: gelöscht ist gelöscht.

4.2.6 Datei->Drucken ...

Dieser Menüpunkt erlaubt das Ausdrucken der Datenfiles, Animations-Skripten und Farb-Paletten. Über den entsprechenden Untermenüpunkt können Sie festlegen, was Sie ausdrucken wollen.

Datenfile: Das aktuelle Datenfile, welches das Aussehen der Objekte bestimmt, wird auf dem in Preferences eingestellten Drucker ausgegeben.

Animations-Skript: Das aktuelle Animations-Skript, das die Bewegungen der Objekte steuert, wird auf dem in Preferences eingestellten Drucker ausgegeben.

Farb-Palette: Die aktuelle Farb-Palette wird auf dem in Preferences eingestellten Drucker ausgegeben.

4.2.7 Datei->Info

Es erscheint ein Kommunikationsfenster, das eine Adresse bekanntgibt, an die Beschreibungen von Fehlern, die vom Anwender gefunden werden, geschickt werden können.

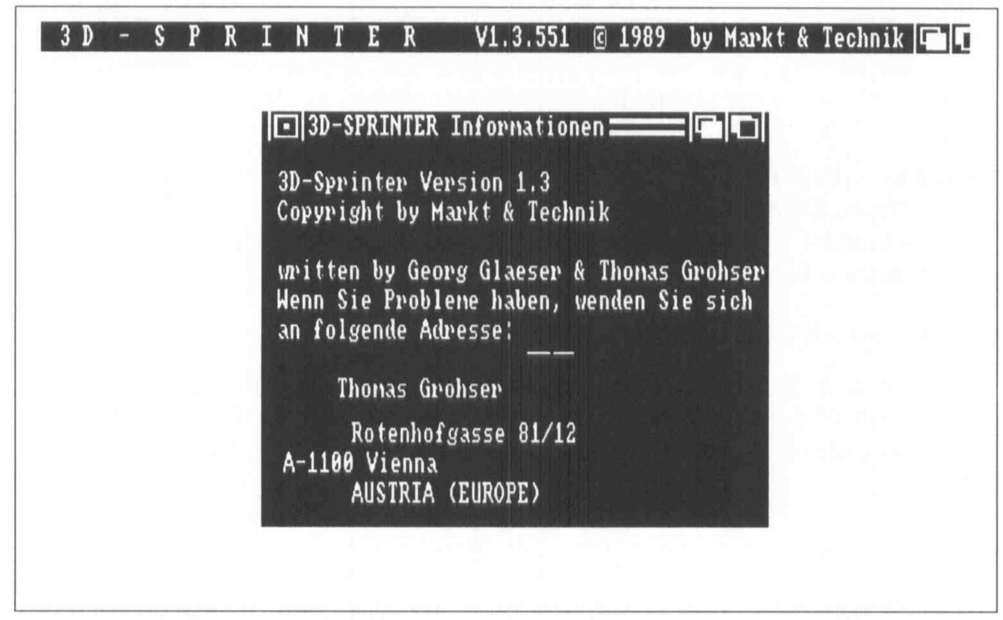


Bild 4.6: Das Info-Kommunikationsfenster

Um das Kommunikationsfenster wieder zu verlassen, genügt ein Klick in das Schließ-Symbol des Fensters.

4.2.8 Datei->Arbeitsumgebung ...

Auch dieser Menüpunkt verbirgt wieder ein Untermenü.

Text-Editor: Um dem Anwender die Benutzung des Programms so einfach wie möglich zu machen, wurde im 3-D-Sprinter kein eigener Texteditor implementiert, statt dessen kann zwischen sechs weitverbreiteten Text-Editoren gewählt werden.

- **SuperEd:** Ein Text-Editor von Markt&Technik
- **Ed:** Der auf jeder Workbench mitgelieferte Text-Editor
- **CygnusEd:** Ein bei Programmierern sehr beliebter Text-Editor
- **MEmacs:** Public Domain-Text-Editor (verschiedenste Versionen auf Public Domain und der Extras-Diskette, die Sie beim Kauf Ihres Amigas bekommen haben)

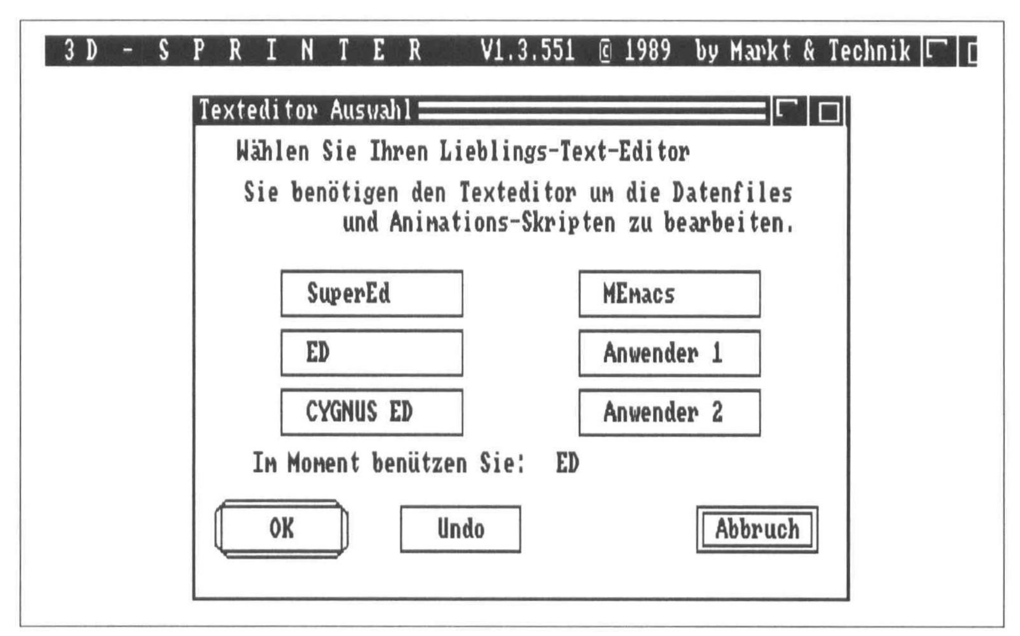


Bild 4.7: Auswahl des Text-Editors

Anwender 1 und Anwender 2: Da wir nicht alle Text-Editoren der Welt kennen und daher auch nicht alle auflisten können, Ihnen aber nicht die Chance nehmen wollen, Ihren Lieblings-Text-Editor zu verwenden, gibt es eine Möglichkeit, jeden beliebigen zu benützen. Dafür ist es allerdings erforderlich, die Datei »3-D-Sprinter.startup« in der Schublade (im Verzeichnis) 3-D-Sprinter zu verändern. Laden Sie diese Datei in »Ihren« Text-Editor. In der dritten und vierten Zeile finden Sie folgenden Text:

```
CUSTOM_1 c:ed
CUSTOM_2 c:vi
```

Statt den Texten c:ed und c:vi können Sie die Editoren angeben, die Ihnen am liebsten sind. Speichern Sie die Datei wieder ab. Von diesem Zeitpunkt an können Sie über die Editoren Anwender 1 und Anwender 2 »Ihren« Text-Editor bzw. Ihre Text-Editoren anwählen.

Dabei ist aber zu beachten, daß der Text-Editor über die Syntax <Name des Text-Editors> <Name der zu editierenden Datei> gestartet wird. Sollte das nicht der Fall sein, setzen Sie sich mit uns in Verbindung, wir werden eine Lösung finden. Aus Urheberrechtsgründen kann nur der »Ed« mitgeliefert werden.

Soll ein anderer Text-Editor als der »Ed« benutzt werden, so muß er zuerst ins »C:«-Verzeichnis der 3-D-Sprinter-Systemdisk kopiert werden (z. B.: von der eigenen Originaldiskette ins »C:«-Verzeichnis der 3-D-Sprinter-Systemdisk kopieren).

Speichern: Alle Einstellungen des »Arbeitsumgebungs«-Menüs werden abgespeichert, um beim nächsten Programmstart gleich wieder zur Verfügung zu stehen.

Zurückholen: Wenn versehentlich Veränderungen an der Arbeitsumgebung (Editor, benutzter Plotter ...) durchgeführt wurden, können mit diesem Menüpunkt die zuletzt abgespeicherten Werte wieder zurückgeholt werden.

Preferences: Das Preferences-Programm von der Workbench wird aufgerufen. Die Bedienung dieses Programms wird in den mit dem Computer ausgelieferten Handbüchern und in Kapitel 1.7 erklärt.

4.2.9 Datei->Ende

Über diesen Menüpunkt können Sie das Programm beenden. Es folgt noch eine Sicherheitsabfrage, bei der man durch Anklicken des linken Symbols »Zurück zum 3-D-Sprinter« wieder ins Programm zurückkehrt. Wird hingegen das rechte Symbol »3-D-Sprinter Verlassen« angewählt, so wird das Programm beendet.



Bild: 4.8: Sicherheitsabfrage, ob das Programm beendet werden soll oder nicht

4.3 Das »Anzeigen«-Menü

Datei **Anzeigen** Edit Animation Animator
Nachricht N

Bild 4.9: Das »Anzeigen«-Menü

Dieses Menü besitzt zur Zeit nur einen Menüpunkt, wird aber zukünftig sicher erweitert werden.

4.3.1 Anzeigen->Nachricht

Ein Aufruf dieses Menüpunktes hat erst nach dem ersten Start des Animators Sinn, da nach der Anwahl dieses Menüpunktes die Fehlermeldungen, Warnungen, Kommentare und Mitteilungen des Animationsteils angezeigt werden. Da diese meist länger als eine Bildschirmseite sind, können Sie mit den Tasten des Ziffernblocks in ihnen blättern, wobei die Tasten folgende Bedeutung haben:

[1]: Textende
[2]: Eine Zeile in Richtung Textende
[3]: Eine Bildschirmseite in Richtung Textende
[7]: Textanfang
[8]: Eine Zeile in Richtung Textanfang
[9]: Eine Bildschirmseite in Richtung Textanfang

Auf den Tastaturen des Amiga 500 und 2000 entspricht die Tastenbelegung der Beschriftung auf der Tasten-Vorderseite. Jede andere Taste beendet das Darstellen der Nachrichten des Animationsprogramms. Der zuletzt ausgegebene Text wird allerdings nicht gelöscht. Dies hat den Vorteil, daß man ihn beim Korrigieren eines Fehlers immer wieder durchlesen kann.

4.4 Das »Edit«-Menü

Über dieses Menü können Sie alle Einstellungen (Licht, Augpunkt, Verdrehungswinkel), die Datenfiles und Animations-Skripten verändern.

Datei Anzeigen Edit Animation Animator

Verdreh-Winkel	T
Augpunkt	E
Lichter ...	
Anzahl der Bilder	B
Datenfile	D
Animations-Skript	F
Farb-Palette	

Bild 4.10: Das »Edit«-Menü

4.4.1 Edit->Verdreh-Winkel

In diesem Kommunikationsfenster können Sie den Startwert für den Verdrehungs-(Twist-)Winkel eingeben.

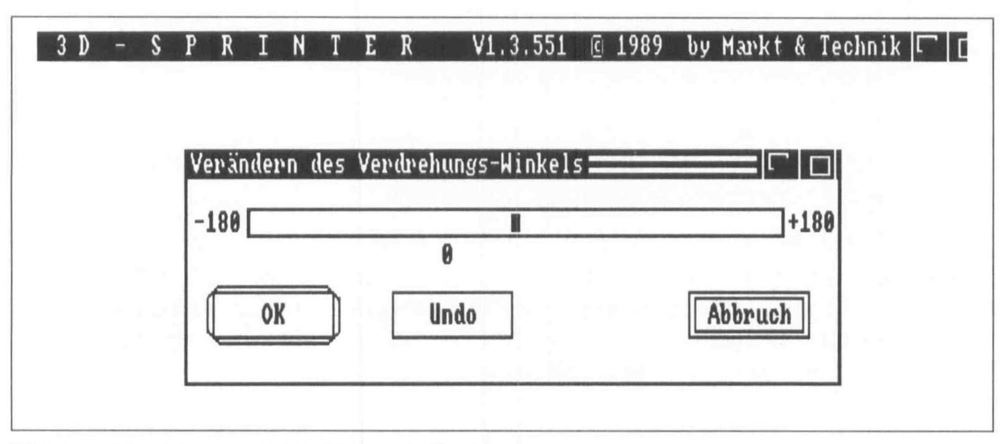


Bild 4.11: Das Verdreh-Winkel-Kommunikationsfenster

Mit dem Schieberegler kann der Wert in Ein-Grad-Schritten im Bereich von –180° bis +180° verändert werden. Ein Klick in das »Undo«-Symbol bewirkt eine Rückstellung des Wertes auf die beim Aufruf des Kommunikationsfensters eingestellte Größe. Mit einem Klick in das »O.K.«-Symbol wird der neu eingestellte Wert übernommen. Das »Abbruch«-Symbol macht alle Änderungen wieder rückgängig und beendet ebenfalls die Eingabe.

4.4.2 Edit->Augpunkt

In diesem Kommunikationsfenster können Sie die Position des Auges (von dem aus die Objekte gesehen werden) verändern. Statt Auge könnten wir auch Kamera bzw. Linse der Kamera sagen.

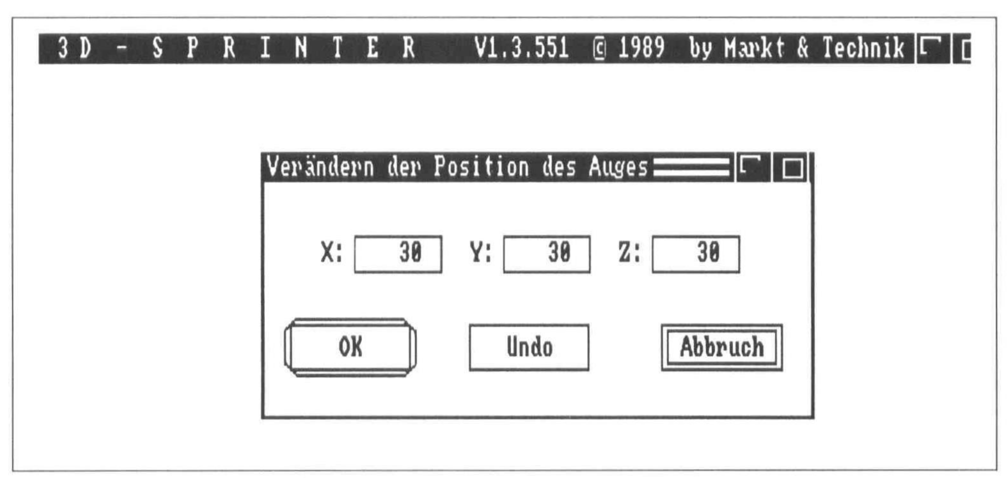


Bild 4.12: Kommunikationsfenster zum Festlegen des Augpunktes

In den drei Text-(Zahlen-)Symbolen in der Mitte des Kommunikationsfensters können Sie Koordinaten im Bereich von –9999 bis +9999 eingeben. Dazu müssen Sie zuerst mit der Maus in das zu verändernde Symbol klicken. Nachdem die Schreibmarke erschienen ist, kann der Wert wie gewohnt verändert werden.

Mit einem Klick in das »Undo«-Symbol bewirken Sie eine Rückstellung der Werte auf die beim Aufruf des Kommunikationsfensters eingestellten Größen. Mit einem Klick in das »O.K.«-Symbol werden die neu eingestellten Werte übernommen. Das »Abbruch«-Symbol macht alle Änderungen wieder rückgängig und beendet ebenfalls die Eingabe.

4.4.3 Edit->Lichter ...

Das Untermenü beinhaltet vier Einträge. Mit den ersten beiden (»1« und »2«) können Sie ein Kommunikationsfenster (gleich dem zum Festlegen der Position des Augpunktes) aktivieren, um die Koordinaten der Lichtquellen einzugeben. Wie dieses zu bedienen ist, lesen Sie bitte in 4.4.2 nach.

Die Menüpunkte mit den Namen »ON« und »OFF« geben an, ob die zweite Lichtquelle ein- oder ausgeschaltet ist. Der aktuelle Zustand des Schalters wird abgehakt. Die erste Lichtquelle ist immer eingeschaltet (sonst bleibt's ja dunkel). Leider hat dieser Menüpunkt in der Version 1.3 noch keine Bedeutung, da nur eine Licht-

quelle zur Verfügung steht, doch in Zukunft wird sicher noch eine zweite Sonne aufgehen.

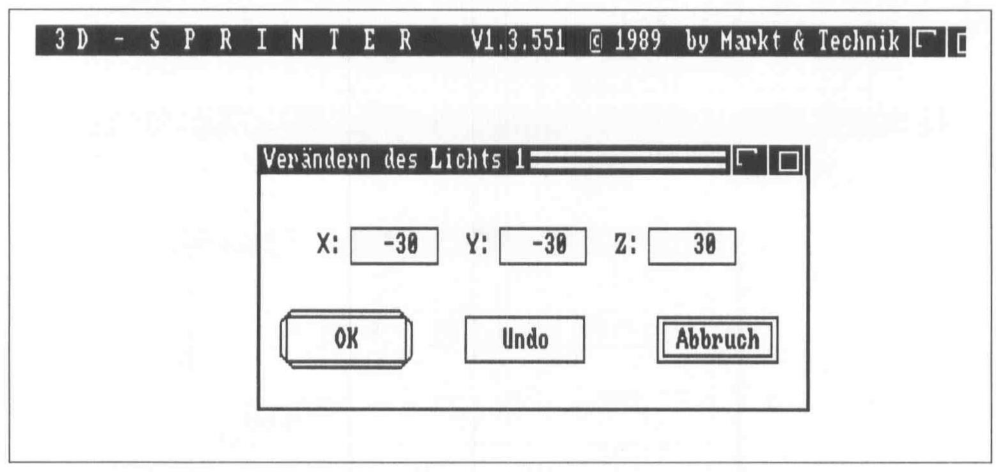


Bild 4.13: Kommunikationsfenster zum Festlegen der Position der Lichtquelle

4.4.4 Edit->Anzahl der Bilder

In diesem Kommunikationsfenster kann die Anzahl der zu berechnenden Bilder eingegeben werden.

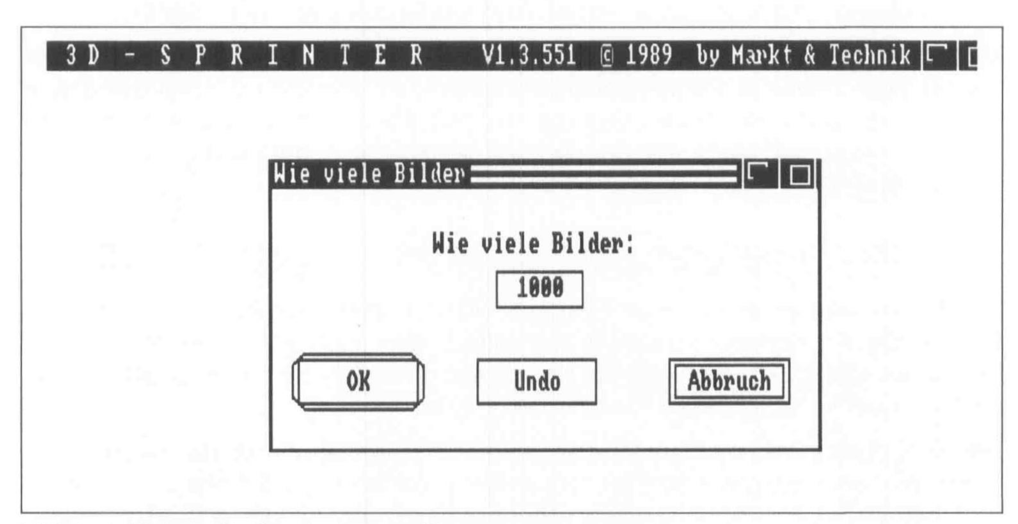


Bild 4.14: Kommunikationsfenster zur Eingabe der Bildanzahl

In dem Text-(Zahlen-)Symbol in der Fenstermitte können Sie Werte von 1 bis 99999 eingeben. Auch dieses Symbol kann wie in 4.4.2 (Edit->Augpunkt) beschrieben, bedient werden. 99999 Bilder klingt nicht viel, doch nehmen wir an, der 3-D-Sprinter benötigt pro Bild eine Sekunde, so würde er an 99999 Bildern mehr als einen Tag rechnen. Da eine aufwendige Szene jedoch ca. 30 Sekunden Rechenzeit benötigt, kann man ihn schon mal für einen Monat beschäftigen (hätte dann allerdings theoretisch bei 20 Bildern pro Sekunde einen Videofilm von eineinhalb Stunden Spieldauer und eine volle Harddisk). Die Symbole »O.K.«, »Abbruch« und »Undo« haben dieselben Funktionen wie bei den vorherigen Kommunikationsfenstern.

4.4.5 Edit->Datenfile

Es wird der über den Menüpunkt »Datei->Arbeitsumgebung ... ->Text-Editor« eingestellte Text-Editor aufgerufen, um das Datenfile, das die Objekte beschreibt, zu editieren.

4.4.6 Edit->Animations-Skript

Es wird der über den Menüpunkt »Datei->Arbeitsumgebung ... ->Text-Editor« eingestellte Text-Editor aufgerufen, um das Animations-Skript, welches die Bewegung der Objekte beschreibt, zu editieren.

4.4.7 Edit->Farb-Palette

Wenn Sie sich gerne selbst jene Farben aussuchen möchten, mit denen Sie Ihre Objekte zeichnen, ist es über diesen Menüpunkt möglich, mit dem im Menüpunkt »Datei->Arbeitsumgebung->Text-Editor« eingestellten Text-Editor die Farb-Palette zu verändern bzw. eine neue aufzubauen.

Um auch den Extra-Halfbright-Modus des Amigas ausnützen zu können, ist das Erzeugen einer Farb-Palette ein wenig kompliziert. Es sind jedoch genügend Beispiele auf der Diskette. Wenn Sie sich zuerst die Beispiele ansehen, werden Sie sicher keine Probleme haben, eigene Farb-Paletten zu erstellen. Hier nun der Aufbau der Farb-Paletten am Beispiel der Standard-Palette:

Jede Palette besteht beim Amiga aus 15 Einzelfarben, die mit den Nummern 1 (dunkelste) bis 15 (hellste) bezeichnet werden.

Zuerst wird der Name der Palette angegeben: grau. Dieser dient nur zur Orientierung, da in den Datenfiles nur die Farben »grau«, »gelb« und »gruen« verwendet werden dürfen. Wobei »grau« den ersten 15 in der Farb-Palette definierten Farben, »gelb« den zweiten 15 und »gruen« den dritten 15 entspricht. Welche Farben dann auf dem Bildschirm zu sehen sind, hängt aber vom Inhalt der Farb-Palette ab.

Danach werden die einzelnen Farben von der hellsten (15) bis zur dunkelsten (1) aufgelistet. Neben der Farbnummer werden die drei RGB-Anteile aufgelistet, und zum Schluß folgt die Beschreibung, woher die Farbe genommen wird.

Der Amiga setzt jede Farbe aus den drei Grundfarben Rot, Grün und Blau zusammen. Die RGB-Anteile geben nun an, aus wievielen Anteilen der drei Grundfarben die neue Farbe zusammengesetzt wird, wobei »0« keinen Anteil bedeutet und 15 vollen Anteil. Demnach bedeutet 0 0 0 Schwarz, 15 15 15 Weiß, 15 0 0 Rot, 0 0 15 Blau und 0 15 0 Grün. Die Angabe, woher die einzelnen Farben genommen werden, ist notwendig, da sonst die Farben des Amiga nicht vollständig ausgenützt werden. Insgesamt gibt es fünf Bezeichnungen für die Entstehungsart einer Farbe.

»new_color«: Es wird eine neue Farbe erzeugt. Insgesamt können 29 neue Farben erzeugt werden, da drei schon vom Programm benötigt werden.

»halfbrite_of nummer«: Wenn sich eine neue Farbe durch Halbierung der RGB-Werte einer anderen Farbe ergibt, kann eine neue Farbe eingespart werden. Geben Sie dann einfach »halfbrite_of« und die Nummer der Farbe, die halbiert werden soll, an. Nun können noch die drei vom Programm benötigten Farben angesprochen werden, und zwar mit »white« für Weiß, »black« für Schwarz und »dark_gray« für Dunkelgrau. Hier nun als erstes Beispiel eine graue Palette:

```
15  15 15 15  white
14  14 14 14  new_color
13  13 13 13  new_color
12  12 12 12  new_color
11  11 11 11  new_color
10  10 10 10  new_color
 9   9  9  9  new_color
 8   8  8  8  new_color
 7   7  7  7  halfbrite_of 14
 6   6  6  6  halfbrite_of 12
 5   5  5  5  halfbrite_of 10
 4   4  4  4  new_color
 3   3  3  3  new_color
 2   2  2  2  halfbrite_of 4
 1   1  1  1  halfbrite_of 3
```

Zum Schluß müssen Sie noch angeben, daß die Palette zu Ende ist: end_gray. Danach können Sie mit der zweiten Palette beginnen:

```
green
   15  13 15 13  new_color
   14  11 14 11  new_color
   13   9 13  9  new_color
   12   8 12  8  new_color
   11   7 11  7  new_color
   10   6 10  6  new_color
    9   5  9  5  new_color
    8   5  7  5  new_color
    7   4  6  4  halfbrite_of 13
    6   3  5  3  halfbrite_of 11
    5   2  4  2  new_color
    4   2  3  2  halfbrite_of 8
    3   1  3  1  new_color
    2   1  2  1  halfbrite_of 5
    1   0  1  0  halfbrite_of 3
end_green
```

Insgesamt sind drei Paletten möglich, deshalb geben wir auch gleich die dritte an:

```
yellow
   15  15 14 11  new_color
   14  15 13  9  new_color
   13  14 12  8  new_color
   12  13 11  8  new_color
   11  12 10  6  new_color
   10  11  9  6  new_color
    9  10  8  5  new_color
    8   9  7  5  new_color
    7   8  6  4  new_color
    6   7  5  4  new_color
    5   6  5  4  halfbrite_of 12
    4   5  4  3  halfbrite_of 10
    3   4  3  2  halfbrite_of 7
    2   3  2  2  halfbrite_of 6
    1   2  2  2  dark_gray
end_yellow
```

Wenn Sie sich die mitgelieferten Beispiele ansehen, werden Sie sicher bald den Aufbau verstanden haben.

4.5 Das »Animation«-Menü

Datei Anzeigen Edit **Animation** Animator
Start ...
Zeichenmodus ...
Distanz-
veränderung ...
Rotation um ...

Bild 4.15: Das »Animation«-Menü

Über dieses Menü legen wir alle Einstellungen über den Ablauf der Animation fest bzw. wird die Animation gestartet.

4.5.1 Animation->Start ...

Über diesen Menüpunkt wird der Animator gestartet. Über das Untermenü wird festgelegt, wie die Animation gesteuert wird:

Interaktiv: Die Objekte können interaktiv über die Tastatur bewegt werden. Wie dies vor sich geht, wurde bereits ausführlich in Kapitel 3 beschrieben. Eine Zusammenfassung aller Möglichkeiten finden Sie im Anhang.

Autorotation: Die Objekte bewegen sich automatisch mit den Einstellungen des Animations-Menüs (siehe weiter unten).

Animations-Skript: Der Animator wird gestartet und alle Bewegungen so ausgeführt, wie es in dem zuvor geladenen oder neu erzeugten Animations-Skript steht. Wie Sie ein Animations-Skript erstellen können, erfahren Sie in Kapitel 5.

Autorot.->Video:
Skript->Video: Wie oben, es wird jedoch vom ersten Bild an ein Video aufgezeichnet, das danach schnell wieder abgespielt werden kann.

Video Replay: Über diesen Menüpunkt können Videos wieder abgespielt werden. Dazu muß man sie entweder gerade vorher erzeugt oder über den Menüpunkt »Datei->Laden->Video« geladen haben. Näheres über das Abspeichern und Abspielen von Videos erfahren Sie in Kapitel 6.

4.5.2 Animation->Zeichenmodus ...

In diesem Menü wird eingestellt, wie die einzelnen Objekte dargestellt werden sollen:

Drahtgitter: Die Objekte werden als Drahtgittermodell ohne Sichtbarkeit gezeichnet. Um die Anschaulichkeit ein wenig zu erhöhen, werden die Linien, die weiter vom Auge entfernt sind, etwas dunkler dargestellt.

Schattiert: Die Objekte werden schattiert gezeichnet. Das bedeutet, daß die Flächen gefüllt werden.

Schlagschatten: Die Objekte werden mit Schlagschatten gezeichnet.

Hidden Lines: Die Objekte werden als Drahtgittermodell mit Sichtbarkeit gezeichnet. Diesen Zeichenmodus sollten Sie wählen, wenn Sie die Zeichnung ausplotten wollen. Der gerade aktive Darstellungsmodus wird durch einen vorangestellten Haken gekennzeichnet.

4.5.3 Animation->Distanzveränderung ...

Über dieses Untermenü wird festgelegt, ob und wie sich die Distanz Auge <-> Objekte verändern soll.

Keine: Es findet keine Änderung statt.

Dist +: Der Abstand Auge <-> Objekte vergrößert sich ständig.

Dist –: Der Abstand Auge <-> Objekte verringert sich ständig. Der aktuelle Schalterzustand wird durch ein vorangestelltes Häkchen dargestellt. Ab einem gewissen Abstand kann dieser nicht mehr verringert werden, da sich sonst das Auge in das Innere der Objekte bewegen würde. Dies wird allerdings vom Programm abgefangen.

4.5.4 Animation->Rotation um ...

In diesem Menü wird festgelegt, ob und wie sich die Objekte drehen.

ON/OFF: Dieser Schalter gibt an, ob überhaupt gedreht wird. Der aktuelle Zustand wird durch einen vorangestellten Haken gekennzeichnet.

Azimutal-Winkel + / 0 / – :

»+« bewirkt eine ständige Erhöhung des Azimutalwinkels.
»0« der Azimutalwinkel wird nicht verändert.
»–« bewirkt eine ständige Erniedrigung des Azimutalwinkels.
Der aktuelle Zustand wird ebenfalls durch Unterlegen gekennzeichnet.

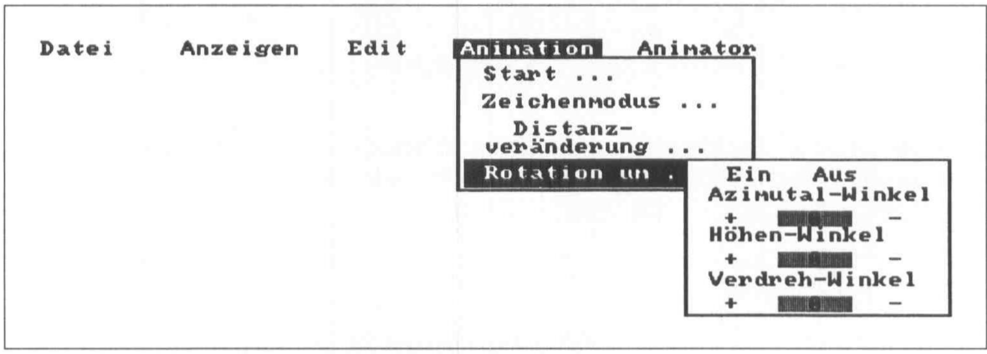


Bild 4.16: Das Untermenü zur Festlegung der Autorotation

Höhen-Winkel + / 0 / – :

Hier gilt sinngemäß dasselbe wie oben.

Verdreh-Winkel + / 0 / – :

Auch hier gilt sinngemäß dasselbe wie oben.

4.6 Das »Animator«-Menü

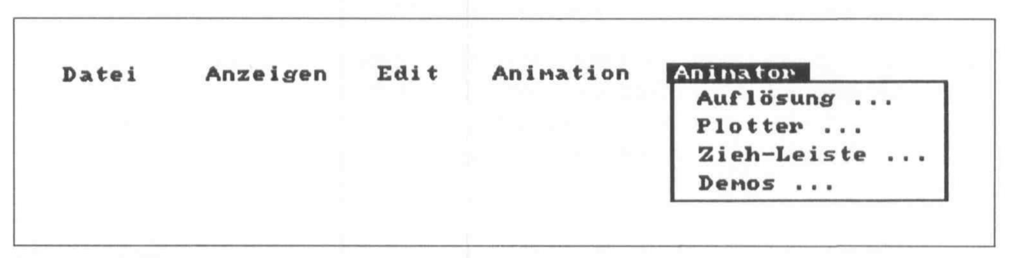


Bild 4.17: Das »Animator«-Menü

In diesem Menü legen wir die Optionen des Animators und die Bildschirmauflösung fest.

4.6.1 Animator->Auflösung ...

In diesem Untermenü wird die Auflösung festgelegt, in der der Animator arbeitet. Durch die Vielzahl der Auflösungen, die der Amiga bietet, ist dies ein sehr umfangreiches Untermenü:

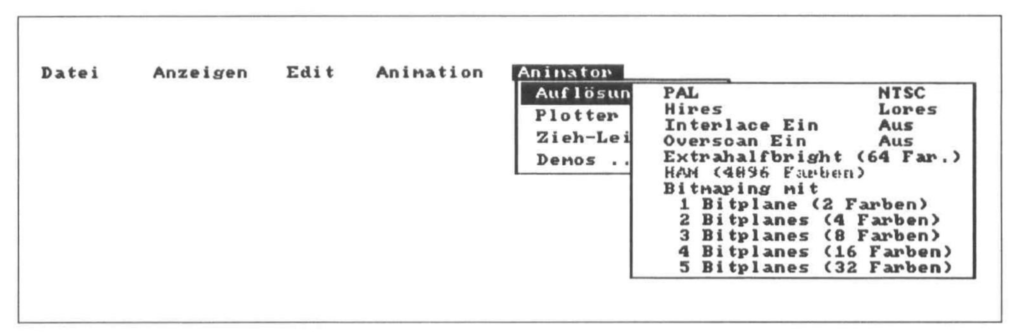


Bild 4.18: Das sind die möglichen Bildschirmauflösungen

Die gerade aktiven Darstellungsmodi werden durch ein vorangestelltes Häkchen gekennzeichnet. Hier eine kurze Erklärung der Darstellungsmodi:

PAL/NTSC: Dies sind die Bezeichnungen für die europäische (PAL) und amerikanische (NTSC) Fernsehnorm. Da es möglich ist, diese abzufragen, werden sie beim Programmstart automatisch richtig gesetzt. Um Ihnen die Möglichkeit zu geben, in den anderen Modus umzustellen, wurde diese Einstellungsmöglichkeit eingebaut.

HiRes/LoRes: Dieser Schalter gibt an, ob der Bildschirm 640 (HiRes, hochauflösend) oder 320 (LoRes, niederauflösend) Punkte (engl.: Pixel = Picture Element) breit ist. Die hochauflösende Darstellung hat allerdings den Nachteil, daß dann nur noch maximal 16 Farben möglich sind.

Interlace Ein/Aus: Durch diesen Schalter kann die vertikale Bildschirmauflösung verdoppelt werden.

Dies hat allerdings zur Folge, daß das Bild zu flimmern beginnt. Das kommt daher, daß nur noch halb so viele Bilder erzeugt werden, wodurch für das menschliche Auge kein stehendes Bild mehr zustande kommt.

Overscan Ein/Aus: Ist für Sie dann interessant, wenn Sie die Animationen mit einem Videorecorder aufnehmen wollen, da der Bildschirm so vergrößert wird, daß die Streifen an den Rändern verschwinden. Sie sollten aber diesen speziellen Modus wirklich nur benützen, wenn Sie mit einem Videorecorder arbeiten, da diese paar Punkte, um die der Bildschirm vergrößert wird, eine Menge Bildschirmspeicher (Chip-RAM) benötigen.

Extra-Halfbright (64 Farben): Dies ist der Darstellungsmodus, in dem der 3-D-Sprinter standardmäßig arbeitet, da in ihm zur Zeit die meisten Farben möglich sind. Doch wie wir das vom Amiga gewöhnt sind, hat auch dieser Modus einen Nachteil. Die Hälfte der Farben ergibt sich aus der anderen Hälfte, indem ihre Helligkeit halbiert wird. Für diese Darstellungstechnik wurden auch die Farb-Paletten (Colormaps) eingeführt. Wie Sie diese erstellen, lesen Sie bitte in 4.4.7

nach. Bei einigen älteren Amigamodellen kann es vorkommen, daß der Extra-Halfbright-Modus nicht funktioniert. In diesem Fall wenden Sie sich bitte an Ihren Händler oder an Commodore direkt (ein neuer Grafikchip [die liebe Denise] bewirkt wahre Wunder).

Ham (4096 Farben): Diese spezielle Darstellungstechnik, bei der alle 4096 Farben, die der Amiga darstellen kann, zugleich auf dem Bildschirm sichtbar sind, ist in der Version 1.3 noch nicht implementiert. Bitmapping mit

1 Bitplane (2 Farben)
2 Bitplanes (4 Farben)
3 Bitplanes (8 Farben)
4 Bitplanes (16 Farben)
5 Bitplanes (32 Farben)

Die Darstellung der Grafik erfogt wie im Extra-Halfbrite-Modus, jedoch mit entsprechend weniger Farben. Dies kann von Vorteil sein, wenn Sie ein Bild im IFF-Standard abspeichern möchten, um es in ein anderes Programm zu übernehmen und dieses Programm z. B. nur mit acht Farben arbeiten kann.

4.6.2 Animator->Plotter

Mit den Plottern verhält es sich wie mit den Text-Editoren. Es gibt eine Vielzahl von Herstellern und Typen. Um wirklich jeden Plotter ansprechen zu können, haben wir uns folgendes System überlegt: Wenn Sie diesen Menüpunkt aufrufen, können Sie mit dem in »Datei-> Arbeitsumgebung ... -> Text-Editor« festgelegten Text-Editor die Einstellungen an Ihren Plotter anpassen. Voreingestellt sind die Werte für einen Epson HI-80:

```
EPSON_HI_80

plot_x_mitte   1255
plot_y_mitte    960
plot_x_max     2510
plot_y_max     1920
plot_stretch      5.5

activate_pen       DF
activate_plotter   IM 31, 11
move_pen           MA %d, %d
draw_pen           DA %d, %d
```

Vorab: Die Reihenfolge der Definitionen darf nicht geändert werden.

Zuerst geben Sie bitte die Bezeichnung Ihres Plotters ein. Achten Sie bitte darauf, daß Sie in dieser Zeile keine Leerzeichen, sondern Unterstriche (_) benützen.

»plot_x_mitte« und »plot_y_mitte« geben die Koordinaten des Mittelpunktes der Zeichnung an. Wie in diesem Beispiel zu sehen, ist es günstig, diesen Mittelpunkt in die Mitte des Blattes zu setzen. Natürlich können Sie ihn auch aus dem Zentrum rücken, wenn Sie die Zeichnung z. B. in der linken oberen Ecke haben möchten.

»plot_x_max« und »plot_y_max« geben die maximalen Koordinaten des Plotters an. Diese Angaben finden Sie sicher in der Beschreibung Ihres Plotters.

»plot_stretch« ist ein Vergrößerungsfaktor, mit dem Sie die Größe der Plotterzeichnung bestimmen können.

Nun folgt die Definition der Befehle, die den Plotter ansteuern:

»activate_pen« ist jener Befehl, der den Schreibstift des Plotters anwählt. Sehen Sie in der Anleitung Ihres Plotters nach, und geben Sie die richtige Zeichenfolge ein.

Nach dem Text »activate_plotter« müssen Sie die Befehle Ihres Plotters angeben, die ihn wieder in den Einschaltzustand zurückversetzen.

»move_pen« ist der Befehl, der den Zeichenstift an eine bestimmte Stelle setzt, ohne dabei zu zeichnen. An den Stellen, an denen die x- und y-Koordinaten eingesetzt werden sollen, schreiben Sie bitte die Zeichenfolgen »%d«.

»draw_pen« ist jener Befehl, der den Plotter dazu veranlaßt, von der Stelle, an die der Zeichenstift mit »move_pen« gestellt wurde, eine Linie zu den neuen Koordinaten zu zeichnen.

Auch hier müssen Sie an den Stellen, an denen die x- und y-Koordinate im Plotterbefehl kommen sollte, jeweils ein »%d« setzen.

Überprüfen Sie bitte noch einmal, ob Sie wirklich bei den Befehlen »move_pen« und »draw_pen« jeweils zwei »%d« stehen haben. Dies ist sehr wichtig!

Als Plotterbefehl wird immer der Bereich ab dem ersten Nicht-Leerzeichen bis zum Ende der Zeile gewertet.

Wenn Sie Schwierigkeiten beim Herausfinden der Plotterbefehle haben, wenden Sie sich an Ihren Händler oder setzen Sie sich mit uns in Verbindung.

Mehr über den Vorgang des Ausplottens von Zeichnungen erfahren Sie in Kapitel 8.

4.6.3 Animator->Zieh-Leiste ...

Um einen Bildschirm mit der Maus bewegen zu können, benötigt er eine Zieh-Leiste. Diese kann aber störend sein, wenn Sie zum Beispiel einen Film erzeugen wollen. Über dieses Menü wird eingestellt, ob die Zieh-Leiste sichtbar (Ein) oder unsichtbar (Aus) ist. Der aktive Zustand wird durch einen vorangestellten Haken gekennzeichnet.

4.6.4 Animator->Demos->Globus

Es wird ein kleines Programm gestartet, das eine rotierende Weltkugel zeichnet.

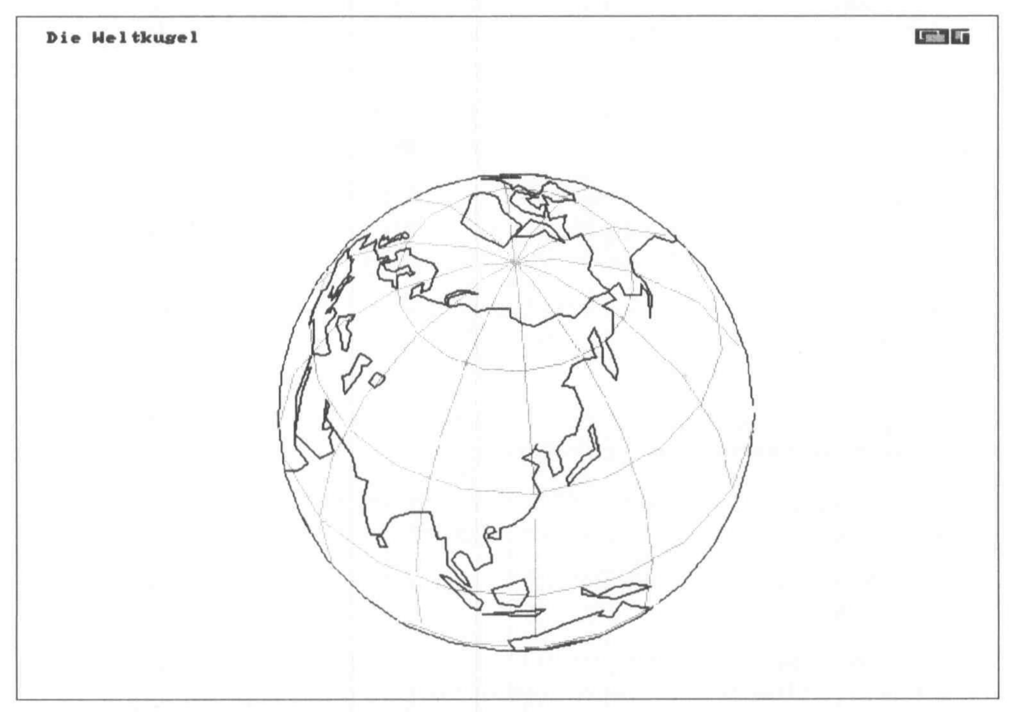


Bild 4.19: Das Globus-Demo

Dieses Demo kann auch über die Tastatur gesteuert werden:

[q] oder [Esc] beendet das Demo.

[a] und [Shift]+ [A] verändern den Azimutal-Winkel.

[e] und [Shift]+ [E] verändern den Höhen-Winkel (Elevation).

[d] und [Shift]+ [D] verändert den Abstand Auge <--> Weltkugel.

[f] schaltet die Autorotation aus.

[Shift]+ [F] schaltet Autorotation wieder ein.

[[] schaltet die Kontinente aus.

[Shift]+ [{] schaltet Kontinente wieder ein.

[]] schaltet die Längen- und Breitenkreise aus.

[Shift]+ [}] schaltet Längen- und Breitenkreise wieder ein.

Mit den Tasten [2], [4], [6] und [8] des Ziffernblocks kann die Position der Weltkugel verändert werden.

4.7 Interaktive Manipulation der Objekte und der Animation

Wenn der Animator im interaktiven Modus gestartet wurde, kann die Szene über die Tastatur bewegt werden. Dies wurde im Kapitel 3 bereits ausführlich erklärt. Eine Zusammenfassung aller Optionen finden Sie im Anhang C. Einige Optionen würden jedoch zu viele Tasten benötigen, deshalb haben wir sie in Kommunikationsfenstern zusammengefaßt.

4.8 Kommunikationsfenster des Animators

Um die Kommunikationsfenster des Animators aufzurufen, genügt es, die [']-Taste zu betätigen (es handelt sich dabei um die Taste links oben über der Tabulator-Taste). Wenn der 3-D-Sprinter das aktuelle Bild fertiggezeichnet hat, öffnet sich in der Bildschirmmitte das folgende Kommunikationsfenster mit dem Hauptauswahlmenü:

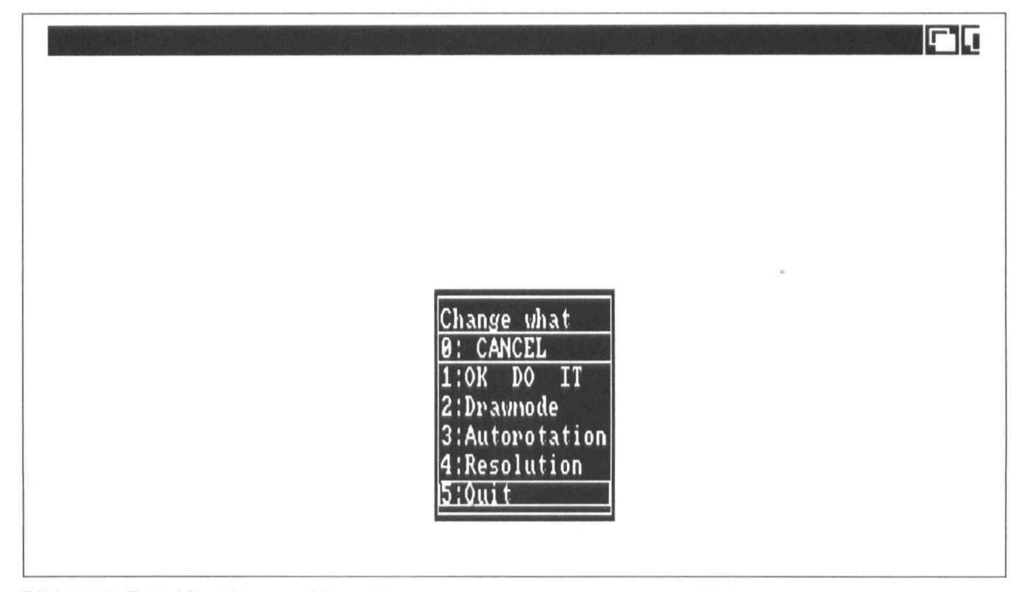


Bild 4.20: Das Hauptauswahlmenü

Wie Sie sehen können, ermöglicht es die Veränderung der Bildschirmauflösung, des Zeichenmodus, der Autorotation und das Verlassen des Programms. Wenn Sie den Menüpunkt »Drawmode« anwählen, gelangen Sie in folgendes Untermenü:

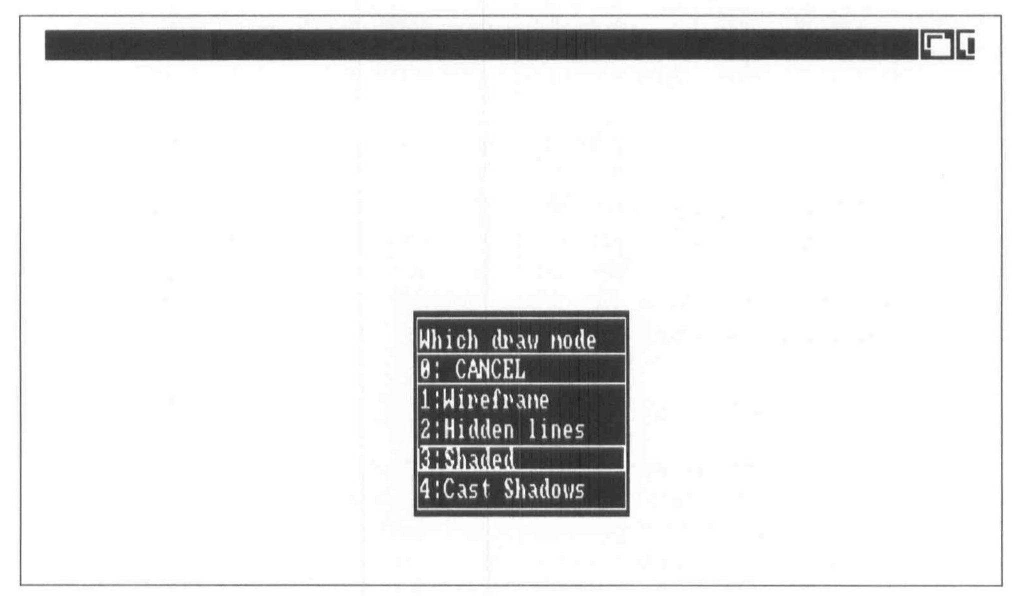


Bild 4.21: Untermenü zum Verändern des Zeichenmodus

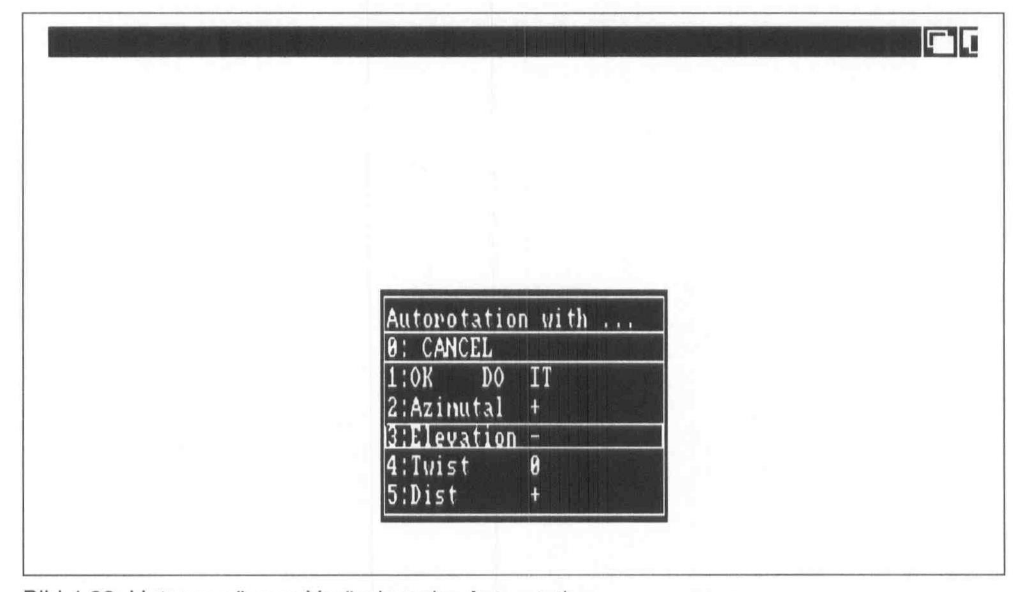


Bild 4.22: Untermenü zum Verändern der Autorotation

Die Möglichkeiten entsprechen genau denen der grafischen Benutzeroberfläche. Über den zweiten Menüpunkt (Bildschirmauflösung) können Sie wie auch über die Benutzeroberfläche den Bildschirm-Modus, mit dem der Animator arbeiten soll, einstellen.

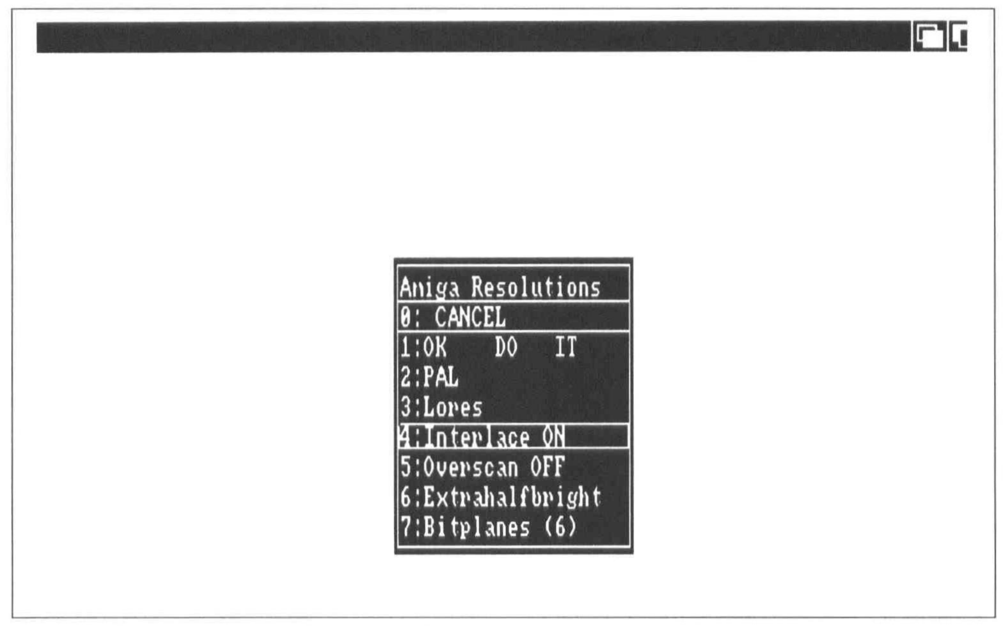


Bild 4.23: Untermenü zum Verändern der Bildschirmauflösung

Mit »O.K. DO IT« werden der oder die Befehle ausgeführt, »CANCEL« bricht unverrichteter Dinge ab.

5 Für Fortgeschrittene: Wir schreiben Animations-Skripten

Trickfilme made by Computer sind heute schon zur Selbstverständlichkeit geworden. Zur – übrigens sehr teuren – Herstellung der diversen Werbespots oder ähnlichen Produktionen werden meist hochspezialisierte Grafik-Workstations verwendet. Der Amiga steht mit seinen Grafikmöglichkeiten bei den Geräten der unteren Preiskategorie weit vorne. Doch ein langer Trickfilm braucht Rechenzeit ... und ein Programmpaket, das in kurzer Zeit jede Menge von realistischen Bildern einer Szene erzeugen kann.

Machen wir einmal eine Hochrechnung: Bei 20 Bildern pro Sekunde brauchen wir pro Minute Filmdauer 1200 Bilder einer Szene. Bei einer Rechenzeit von 30 Sekunden für eine mittelkomplizierte Szene mit Schatten sind das zehn Stunden reine Rechenzeit pro Minute Trickfilm. Also eine Nacht. Und am Abend vorher haben Sie das Animations-Skriptum dazu geschrieben. So gesehen sind das Zeiten, die so manchen Trickfilmhersteller aufhorchen lassen sollten (man munkelt, daß manchmal die Herstellung eines einzigen Bildes so lange dauern kann). Animations-Skripten werden ähnlich wie Datenfiles in »3-D« (siehe dazu Kapitel 7.1) geschrieben. Der Aufbau ist denkbar einfach. Schreiben wir daher gleich ein einfaches Skriptum.

5.1 Erstellen des Skriptums

Begonnen wird, indem die Gesamtanzahl der Einzelbilder angegeben wird, um nötige Dimensionierungen vornehmen zu können (fangen wir bescheiden mit 30 Bildern an):

```
insgesamt 30 bilder
```

Die Wörter »insgesamt« und »bilder« sind Schlüsselwörter, ohne die der Animator abbricht.

Nun erwartet das Programm Informationen über den Augpunkt, das Licht und den Vergrößerungsfaktor beim ersten Bild:

```
anfang
augpunkt 50 -20 30
licht 100 -50 100
vergroesserung 1.2
```

Keine der Angaben darf fehlen, wenngleich die Reihenfolge unbedeutend ist. Wenn Sie sonst nichts dazuschreiben, wird automatisch ein Drahtgittermodell erzeugt, der Koordinatenursprung fixiert und der Verdrehungswinkel auf Null gesetzt. Sie können das aber auch ändern, wenn Sie wahlweise (und in beliebiger Reihenfolge) Zeilen der folgenden Art dazuschreiben:

```
verdrehung 20
fixiere 2 4 5    /* 'fixiere zentrum' ist nicht erlaubt */
schatten         /* bzw. schattiert bzw. verdeckte_linien */
```

Nun wollen Sie natürlich nicht für alle Bilder einzeln Daten hinschreiben müssen (abgesehen davon ergäbe das nur schwer einen kontinuierlichen Bewegungsfluß). Sie wollen aber, daß bei Bild 10 der Augpunkt nach (40, –25, 25) gelangt ist (alles andere soll gleich bleiben). Schreiben Sie's hin!

```
bild 10
   augpunkt 40 -25 25
```

Von Bild 1 bis Bild 10 blieb der Darstellungsmodus auf »schatten«, der Verdrehungswinkel auf 20 Grad, die Vergrößerung 1.2 usw. Der Augpunkt ist kontinuierlich von (50, –20, 30) nach (40, –25, 25) gewandert. Wie diese Wanderung exakt vor sich gegangen ist, läßt sich erst sagen, wenn wir das Datenfile weiterverfolgen. Sagen wir, der Augpunkt soll bei Bild 20 nach (30, –32, 22) und bei Bild 30, also am Ende der Animation, nach (25, –37, 19) gelangt sein:

```
bild 20
   augpunkt 30 -32 22
ende
   augpunkt 25 -37 19
```

Der 3-D-Sprinter liest nun vor dem Einschalten der Grafik das gesamte Animations-Skriptum durch, merkt sich die Zwischenlagen des Augpunktes und berechnet über Spline-Interpolationen (siehe dazu Kapitel 7.6) für jedes Bild die Zwischenlage des Augpunktes. Dadurch ergibt sich ein flüssiger Ablauf der Bewegung, und es treten keine ruckartigen Veränderungen wie bei linearen Interpolationen auf.

Dasselbe würde der 3-D-Sprinter mit dem Lichtzentrum machen (wenn wir veränderte Lichtpunktkoordinaten hingeschrieben hätten), und genauso hätte er den Vergrößerungsfaktor, den Verdrehungswinkel und jenen Punkt, der auf der Linsenachse liegen soll (den »fixierten Punkt«), interpoliert. Die Darstellungsart (drahtgitter, schattiert, schatten, verdeckte_linien) bleibt so lange unverändert, bis Sie etwas anderes ins Skriptum schreiben.

Starten Sie jetzt bitte das entsprechende Datenfile mittels Animations-Skriptum (wie das genau geht, steht in Kapitel 4.5.1). Weil Sie als Darstellungsart »schatten« gewählt haben, dauert Ihnen die Animation beim Testen zu lange. Brechen Sie also wie gewohnt mit der [q]-Taste (oder mit [Esc]) ab, schreiben Sie »drahtgitter« statt »schatten« ins Skriptum und starten Sie erneut. Ja, diesmal

geht's wesentlich schneller. Hier und da noch eine kleine Änderung beim Augpunkt, vielleicht auch beim fixierten Punkt, fertig.

Wenn Sie das Licht auch verändert haben, müssen Sie auch dieses testen. Wenn die Berechnung langatmig ist, machen Sie einfach folgendes: Schreiben Sie – sagen wir bei jedem zehnten Bild – »schatten« vor, und schalten Sie aber gleich wieder in den Drahtgittermodus:

```
bild 10 schatten  /* und andere Veränderungen */
bild 11 drahtgitter
bild 20 schatten  /* und andere Veränderungen */
bild 21 drahtgitter
```

So können Sie mit einem Minimum an Zeitaufwand das Skriptum testen, bevor Sie den endgültigen Ablauf der Animation beginnen. Es ist jetzt 22 Uhr und morgen früh wollen Sie eine Minute Trickfilm auf Ihrer Harddisk gespeichert haben (Sie haben doch dort 10 Mbyte Speicherplatz frei, oder?). Dann werden Sie das Abgespeicherte mit dem »replay«-Programm abspielen und auf Video aufnehmen.

Ach ja, eines haben Sie fast vergessen: Ihr Auftraggeber will auch eine Art »comic strip« von ihrem Trickfilm, sozusagen einen Trickfilm auf dem Papier. Also tauschen Sie das Wort »schatten« durch »verdeckte_linien« aus, starten Sie die Animation, und drücken Sie jedesmal, wenn Sie ein Plotterbild haben wollen, auf [p]. Wenn Ihnen das zu kompliziert ist, geht's auch mit [c] wie »comic strip«, und der Plotter zeichnet die einzelnen Bilder in der Animation automatisch so, daß nach 30 Bildern das Zeichenblatt optimal ausgenützt ist. Weil – wie schon einmal gesagt – Bilder mehr als Worte sagen, gleich ein paar Illustrationen zu dem Gesagten.

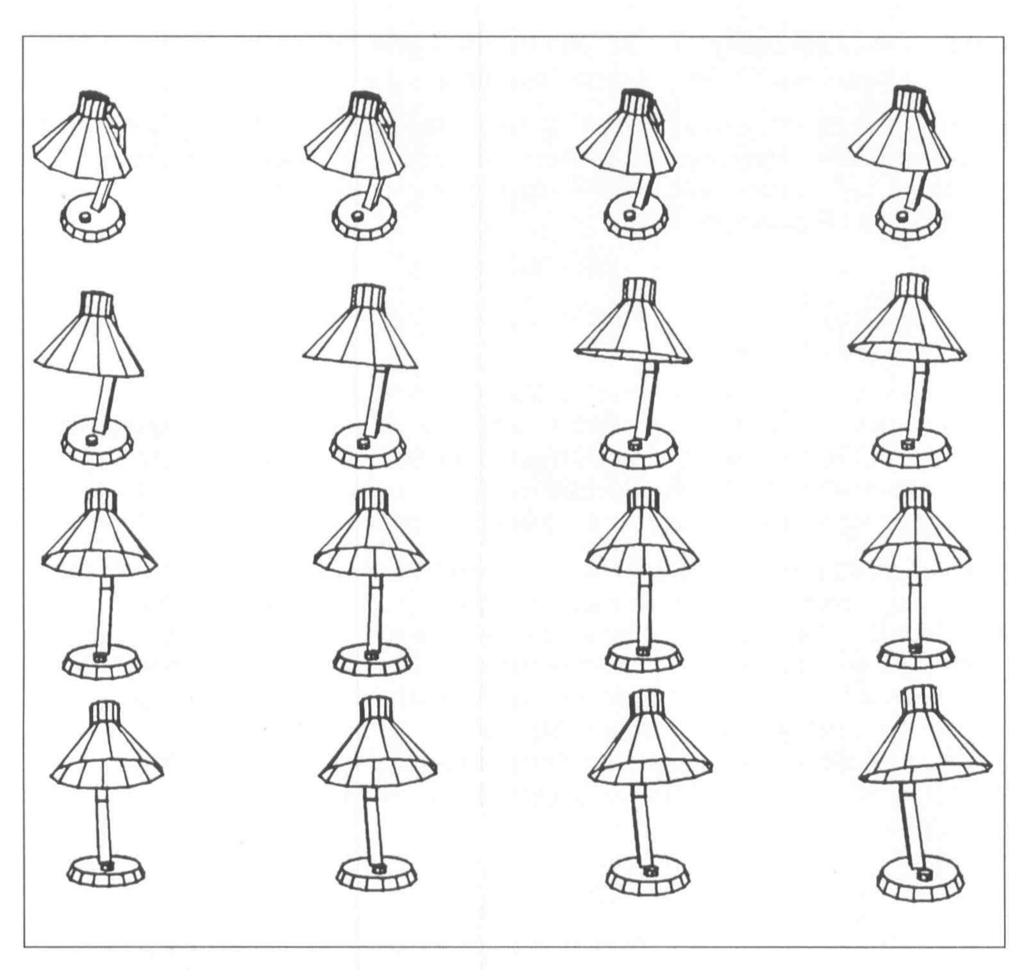


Bild 5.1: Animation der Tischlampe

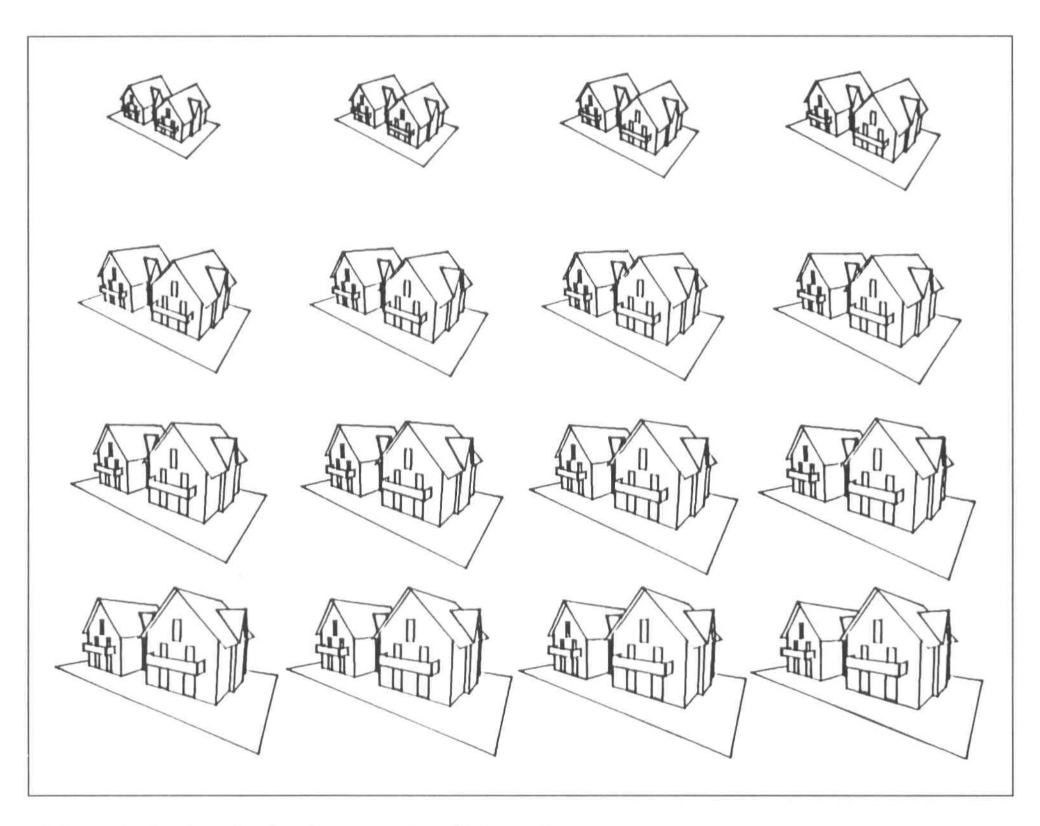


Bild 5.2: Hubschrauberlandung vor den Reihenhäusern

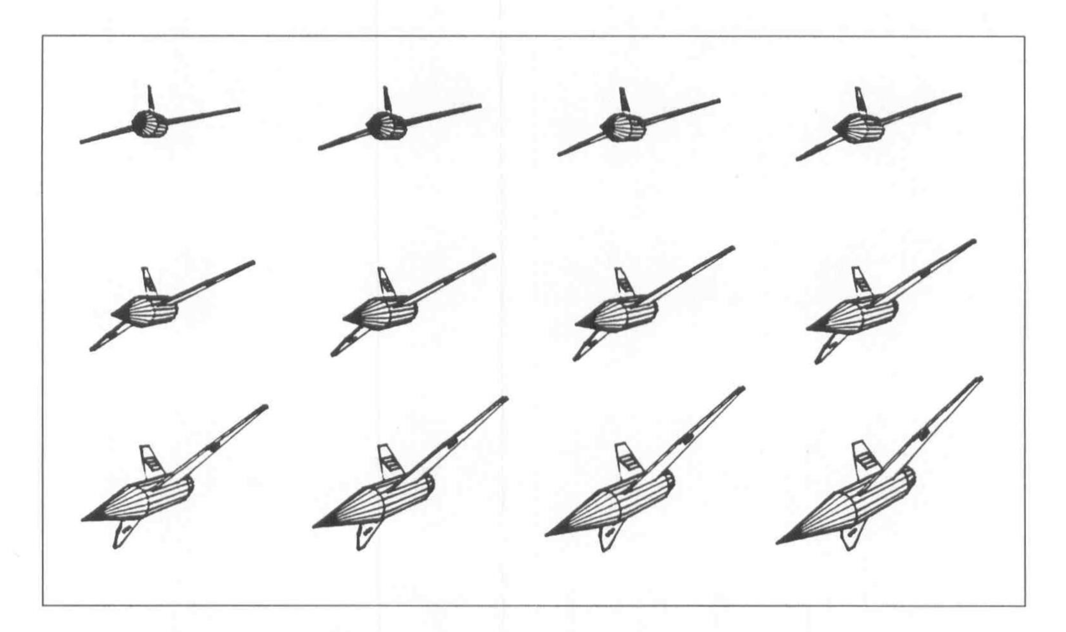


Bild 5.3: Begegnung über den Wolken

5.2 Benutzen des Animations-Skripts

5.2.1 Über die Benutzeroberfläche

Das Animations-Skript ist fertig – also starten wir es über die grafische Benutzeroberfläche: Der Menü-Punkt »Animation->Start ...->Animations-Skript« ist dafür zuständig. Wer zwar die Maus süß findet, aber mit der Tastatur schneller arbeiten kann (soll vorkommen), drückt einfach die Tasten-Kombination [RAmiga]+ [s].

5.2.2 Im CLI

Wenn Sie im CLI bzw. in der Shell arbeiten, hängen Sie bitte die Option »-a <Name des Animations-Skripts>« an den Befehlsaufruf an, also zum Beispiel:

```
try  flugzeug.data -a anim_scripts/flugzeug.anim
```

Wichtig ist dabei, daß Sie zumindest beim Animations-Skriptum den vollständigen Pfad angeben.

5.3 Hinweise für Anfänger beim Erstellen von Trickfilmen

1) Wenn nicht alles nach Wunsch zu gehen scheint, sehen Sie sich die »Nachricht« (Menüpunkt »Anzeigen->Nachricht« oder kürzer [RAmiga]+ [n]) an. Dort sollte eine Meldung stehen, die Ihnen weiterhilft, wie zum Beispiel: »Bildnummern müssen aufsteigend sortiert sein!«
2) Zu große Sprünge bei den Augpunktkoordinaten (Lichtkoordinaten, Vergrößerungsfaktoren usw.) wirken bei einem Trickfilm unelegant. Zu kleine Sprünge bringen die Zuschauer allerdings bald zum Gähnen ...
3) Es ist noch kein Meister vom Himmel gefallen.

6 Das Abspeichern und Abspielen von Animationen-»Videos«

6.1 Warum speichern wir Videos eigentlich ab ?

Eine berechtigte Frage, da der 3-D-Sprinter die einzelnen Bilder fast in Echtzeit berechnet. Dennoch kann es manchmal notwendig sein, eine Animation noch schneller abzuspielen. Ein weiterer Vorteil der Videos ist, daß sie weitergegeben werden dürfen. Dies resultiert aus der Tatsache, daß das »Replay«-Programm (und nur das »Replay«-Programm) beliebig weitergegeben werden darf. Dadurch können Sie die von Ihnen erzeugten Animationen Ihren Freunden und Bekannten zugänglich machen.

6.2 Den Namen des Videos festlegen

Bevor ein Video erzeugt werden kann, muß sein Name bekannt sein, da das Video nicht zuerst im Speicher erstellt wird, sondern direkt auf der Diskette oder Festplatte. Von der Benutzeroberfläche aus können Sie über den Menüpunkt »Datei-> Speichern als ...->Video« den Namen des Videos angeben. Es öffnet sich hierzu das bekannte Dateiauswahl-Kommunikationsfenster. Geben Sie einen neuen Namen ein oder wählen Sie ein bereits vorhandenes Video aus, das Sie überspielen wollen. Wie der Name von CLI aus festgelegt wird, erfahren Sie gegen Ende des nächsten Kapitels.

6.3 Starten der Aufzeichnung

Nachdem Sie den Namen des Videos festgelegt haben, können Sie mit der Aufzeichnung beginnen, dabei müssen Sie allerdings zwei Fälle unterscheiden:

a) Sie wollen die Animation vom ersten Bild an aufnehmen. Dies ist vermutlich in Verbindung mit Animations-Skripten besonders interessant.
b) Sie wollen mitten in der Animation mit der Aufzeichnung beginnen.

Für den Fall a) sind die Menüpunkte »Animation->Start ...->Autorot. -> Video« und »Animation->Start ...->Skript -> Video« zuständig.

Wird der Animator über einen der beiden Menüpunkte gestartet, wird das Video vom ersten Bild an abgespeichert. Wenn Sie hingegen erst mitten in der Animation

mit der Aufzeichnung beginnen wollen – »Fall b)« –, starten Sie den Animator wie gewohnt. Um dann mit der Aufzeichnung zu beginnen, genügt es, die [v]-Taste zu drücken.

Für beide Fälle gilt, daß der Aufzeichnungsvorgang jederzeit durch die Tastenkombination [Shift]+ [V] beendet werden kann. Ebenfalls wird die Aufzeichnung natürlich beim Verlassen des Animators durch [q] oder [Esc] bzw. nach der Abarbeitung eines Animations-Skripts beendet.

6.4 Wiederabspielen des Videos

Doch was nützt uns das schönste Video, wenn wir es uns nicht öfter ansehen können. Aber auch dafür können wir eine Lösung anbieten: Das »Replay«-Programm. Dieses kleine Programm, welches Sie auch weitergeben dürfen, kann die vom 3-D-Sprinter erzeugten Videos abspielen. Es kann von der Benutzeroberfläche aus aufgerufen werden.

6.4.1 Wiederabspielen von der Benutzeroberfläche aus

Wenn Sie das Video gerade erzeugt haben, können Sie es sofort wieder abspielen, wenn nicht, müssen Sie es zuerst über den Menüpunkt »Datei->Laden->Video« laden. Der Abspielvorgang wird über den Menüpunkt »Animation->Start ...->Replay Video« gestartet. Der Arbeits-Schirm wird geschlossen, der Mauszeiger nimmt wieder sein normales Aussehen an und das »Replay«-Programm beginnt mit dem Laden der Animation. Nach dem Laden der Animation beginnt das Programm mit dem Abspielen derselben. Der Programmablauf kann wie gewohnt durch die [Esc]- oder [q]-Taste abgebrochen werden. Mit der [r]-Taste kann die Abspielrichtung (vorwärts/rückwärts) gewechselt werden. Wenn die Animation komplett abgespielt ist, wird sie erneut abgespielt, jedoch in der entgegengesetzten Richtung.

6.4.2 Wiederabspielen vom CLI aus

Die korrekte Befehlssyntax zum Starten des Programms vom CLI oder von der Shell aus lautet:

```
replay <video>
```

wobei <video> das abzuspielende Video ist.
Hier noch ein Beispiel:

```
replay df1:video/poker.video
```

würde ein Video abspielen, das sich auf einer Diskette im Laufwerk »DF1:« in der Schublade »video« befindet und den Namen »poker.video« trägt. Auch in diesem

Fall wird die Animation mit der [Esc]- bzw. [q]-Taste beendet und die Abspielrichtung mit der [r]-Taste verändert.

6.4.3 Wiederabspielen von der Workbench aus

Um ein Video von der Workbench aus abspielen zu können, ist ein wenig Vorarbeit notwendig. Einen Großteil davon haben wir schon für Sie erledigt, denn auf der Daten-Diskette, die dem Buch beiliegt, sind viele Beispiele enthalten, die Sie einfach kopieren und anschließend für Ihre eigenen Anforderungen benutzen können.

Das Abspielen von der Workbench aus wird durch das Programm IconX, das sich auf der Worbench 1.3 (und natürlich auch auf der 3-D-Sprinter-Programm-Diskette) befindet, möglich.

Um ein Programm von der Workbench aus starten zu können, benötigt es ein Piktogramm (Icon). Wir haben eines entworfen und es auf die Datendiskette kopiert. Es trägt den Namen »Video_Muster«.

Im selben Verzeichnis muß sich eine Text-Datei mit demselben Namen befinden. In dieser Text-Datei können beliebige CLI-Befehle stehen. Diese werden dann ausgeführt, wenn das zugehörige Piktogramm mit einem Doppel-Klick aktiviert wird.

Das Piktogramm muß aber vom richtigen Typ sein. Am einfachsten ist es, das von uns mitgelieferte Piktogramm (»Video_Muster«) über den Menüpunkt »Workbench-> Duplicate« der Workbench zu kopieren und anschließend zu modifizieren. Dies geht folgendermaßen vor sich:

Kopieren Sie das Piktogramm (»Workbench->Duplicate« im Menü der Workbench).
Geben Sie dem neuen Piktogramm einen anderen Namen (»Workbench-> Rename«).
Editieren Sie die Text-Datei mit einem Text-Editor.

Die Original-Datei sieht folgendermaßen aus:

```
.key ""
echo "Ich lade die Animation NAME_DES_VIDEOS in den Speicher."
echo "Bitte haben Sie Geduld ...*N"

replay NAME_DES_VIDEOS

type t:message
ask "*N   RETURN um das Programm zu beenden"
```

Sie müssen lediglich statt den Platzhaltern NAME_DES_VIDEOS den richtigen Text einsetzen. Speichern Sie die modifizierte Text-Datei wieder ab. Von nun an können Sie das besagte Video, wenn es bereits existiert, durch einen Doppelklick abspielen.

6.5 Zusammenschneiden mehrerer Videos zu einem Film

Wenn Sie einen Film aus mehreren einzelnen Videos zusammenschneiden wollen, benötigen Sie das Programm »cut«. Dieses fügt mehrere Videos zu einem Film zusammen. Diese Option werden Sie auch benötigen, wenn Sie in einer Animation die Lage der Einzelobjekte oder deren Form verändern wollen, da der 3-D-Sprinter dies zur Zeit noch nicht zuläßt.

Doch nun zum Programm »cut«. Es besitzt einen kleinen Nachteil: Da wir die Idee zu diesem Utility erst einige Tage vor dem Abgabetermin der Programmdiskette an den Verlag hatten, kann es vorläufig nur vom CLI aus benutzt werden. Weil wir es Ihnen trotzdem nicht vorenthalten wollten, haben wir uns entschieden, es »nur« als CLI-Kommando zur Verfügung zu stellen. Hier nun die Befehls-Syntax:

```
cut <Ziel_Video> <Video_1> <Video_2> [<Video_3> ...]
```

Dabei ist <Ziel_Video> der Name des neuen Videos, und <Video_1> bis <Video_x> sind die Namen der bereits vorhandenen Videos.

Beachten Sie bitte, daß Sie die Videos alle mit der gleichen Bildschirmauflösung aufnehmen (bzw. aufgenommen haben), da sonst das »cut«-Programm mit einer Fehlermeldung abbricht. Auch sollten die verwendeten Farben in allen Videos gleich sein. Und hier ein Beispiel:

```
cut df1:kaffee video/kaffee_zu video/kaffee_halb video/kaffee_auf
```

erzeugt aus den Videos »kaffee_zu kaffee_halb kaffee_auf« aus dem Verzeichnis video, das sich im aktuellen Verzeichnis befindet, auf der Diskette in Laufwerk »df1:« das Video »Kaffee«.

6.6 Abspielen von langen Videos

Ein Video kann schnell länger als der vorhandene Hauptspeicher sein, deshalb ermöglicht es eine modifizierte Version des »Replay«-Programms (»HDReplay«), Videos direkt von der Diskette oder der Festplatte (= HardDisk und daraus HDReplay) abzuspielen.

Dies geht jedoch etwas langsamer als das »Replay«-Programm, und die Animation kann nur in eine Richtung abgespielt werden. Auch dieses Programm kann, wie in Kapitel 6.4.3 beschrieben, von der Workbench aus gestartet werden. Verwenden Sie als Vorlage das Piktogramm »HD_Video_Muster« auf der Datendisk.

Von der Benutzeroberfläche aus wird immer das »Replay«-Programm aufgerufen. Sie können aber, wenn Sie wollen, das »Replay«-Programm in »MEMReplay« und das »HDReplay« in »Replay« umbenennen. Danach benutzt die grafische Benutzeroberfäche das »HDReplay«-Programm.

7 Mehr Informationen über das Erstellen von Datenfiles

Der 3-D-Sprinter hat eine Vielzahl von Möglichkeiten eingebaut. Es sollen diese in diesem Kapitel direkt an Hand von Beispielen erarbeitet werden. Dazu vorher ein paar grundsätzliche Worte:

7.1 Die Programmiersprache »3-D«

Wenn wir ein Programm in einer höheren Programmiersprache (C, Pascal usw.) schreiben, müssen wir uns strikt an gewisse Regeln halten. Jede Programmiersprache kennt eine Vielzahl von reservierten Wörtern (»Schlüsselwörter«), auf die der Compiler reagiert, um den Code in Maschinensprache zu übersetzen.

Je systematischer man das Programm editiert (Einhalten von Abständen, sinnvolle Namensgebung für die Variablen), desto leichter ist es, sich im File zurechtzufinden, Informationen daraus abzulesen und Fehler zu finden.

Das Gesagte gilt nun ohne Einschränkung für Ihre Datenfiles, die Sie dem 3-D-Sprinter (Ihrem »Compiler«) zur Verfügung stellen. Der 3-D-Sprinter liest das File Wort für Wort ein und versucht, bei Schlüsselwörtern geeignete Unterprogramme anzuspringen. Wenn dann in der richtigen Reihenfolge weitere Schlüsselwörter folgen, reagiert er neuerlich darauf usw. So gesehen könnte man von einer selbstentwickelten (und noch sehr ausbaufähigen) Programmiersprache reden, der wir spaßeshalber den Namen »3-D« gegeben haben.

Die Schlüsselwörter wurden so gewählt, daß sie wie Kommentare wirken und Sinn ergeben. Mit einiger Übung brauchen Sie nach einer gewissen Zeit kaum noch Kommentarzeilen in Ihre Files zu schreiben.

Versuchen Sie bitte von Anfang an beim »Programmieren in 3-D« dieselben Regeln einzuhalten, wie Sie dies auch beim Programmieren in anderen höheren Computersprachen machen:

- ❑ Benennen Sie Ihre Objekte sinnvoll.
- ❑ Schreiben Sie die Files strukturiert, indem Sie im Text einrücken, Abstände lassen usw.

- Kommentieren Sie schwierigere Passagen.
- Schlüsselwörter werden immer klein geschrieben. Schreiben Sie zur besseren Unterscheidung eigene Namen daher groß (zumindest den Anfangsbuchstaben). Das ist aber nicht zwingend nötig.

7.2 Manipulationen der Bausteine

Die Leistungsfähigkeit von »3-D« wird nicht zuletzt dadurch erreicht, daß man mit wenigen Zeilen Transformationen auf einzelne Bausteine oder auch ganze Bausteingruppen ausüben kann. Die drei prinzipiell verschiedenen Manipulationen sind:

7.2.1 Translationen

Durch Angabe eines Schubvektors nach dem Schlüsselwort »translation« wird der vorangegangene Baustein (bzw. die vorangegangene Bausteingruppe) parallelverschoben. Beispiele für die verschiedenen Möglichkeiten sind:

```
translation 3.2 -5.6 4
translation x 7
translation z -3.0
```

7.2.2 Rotationen

Durch Angabe eines Winkels (im Gradmaß) nach dem Schlüsselwort rotation wird der vorangegangene Baustein (bzw. die vorangegangene Bausteingruppe) verdreht. Beispiele dafür sind

```
rotation y 30.0            /* um die y-Achse um 30 Grad */
rotation z -90             /* um die z-Achse um -90 Grad */
rotation euler -90 30 90 /* »Euler - Winkel«:
                              gleichwertig mit den Zeilen
                              rotation z -90
                              rotation x  30
                              rotation z  90   */
```

Das letzte Beispiel bedarf einiger Erklärung: Folgen dem Schlüsselwort »rotation« das Wort »euler« und drei Winkel, so werden diese Winkel im Sinne von EULER interpretiert. Das Ergebnis ist zugegebenermaßen nur nach einiger Übung leicht vorherzusagen, so daß eine Aufsplitterung in drei Zeilen vernünftiger erscheint. Für Denksportler: Die oben angeführten Eulerwinkel bewirken eine simple Drehung um die y-Achse um 30 Grad. Überhaupt können beliebig viele Drehungen hintereinander stets durch eine einzige ersetzt werden. Dementsprechend ist es auch erlaubt, eine beliebige Drehachse durch einen Punkt auf der Achse und die Achsenrichtung vorzuschreiben:

```
rotation achsenpunkt 5 0 0 achsenrichtung 3 2 0 winkel 35
```

Diese Schreibweise hat den Vorteil, daß auch um Achsen verdreht werden kann, die nicht durch den Nullpunkt gehen. Ein typisches Beispiel dafür ist wieder einmal die Kaffeekanne, wo der Deckel um eine zur x-Achse parallele Gerade gedreht wird. Erlaubt sind die Abkürzungen

```
achsenrichtung x   /* statt Achsenrichtung 1 0 0 */
achsenrichtung y   /* 0 1 0 */
achsenrichtung z   /* 0 0 1 */
```

7.2.3 Streckungen

Ein nützliches Schlüsselwort ist das Wort »streckung«. Dadurch können Bausteine oder ganze Bausteingruppen vergrößert oder verkleinert werden. Beispiel:

```
    gruppe BAUM1 {
   ...
   ...
}
gruppe BAUM2 wie BAUM1
   translation x 7
   streckung zentrum 0 0 -3 faktor 0.7
```

Das Streckzentrum und der Streckfaktor werden wie in der letzten Datenzeile angegeben. Die drei Manipulationsarten können nun in beliebiger Reihenfolge hintereinander ausgeführt werden. Dabei kann jede Stellung im Raum erreicht werden, wenn man systematisch vorgeht. Beispiele sind in den folgenden Listings reichlich zu finden. Halten Sie sich bitte immer wieder vor Augen, daß es nicht gleichgültig ist, in welcher Reihenfolge die Drehungen bzw. Translationen aufeinander folgen!

7.3 Aufmalen von Mustern

Viele der angegebenen Beispiele haben auf größeren Flächen Muster aufgemalt. Das klassische Beispiel sind die Pokerwürfel. Wie kann man das nun bewerkstelligen? Nehmen wir als Beispiel einen Drehkörper, etwa einen zwölfseitigen Pyramidenstumpf. Bild 7.1 zeigt, wie der 3-D-Sprinter die Flächen numeriert. Auf Fläche Nummer 3 wollen wir den Buchstaben »i« in grüner Farbe aufmalen.

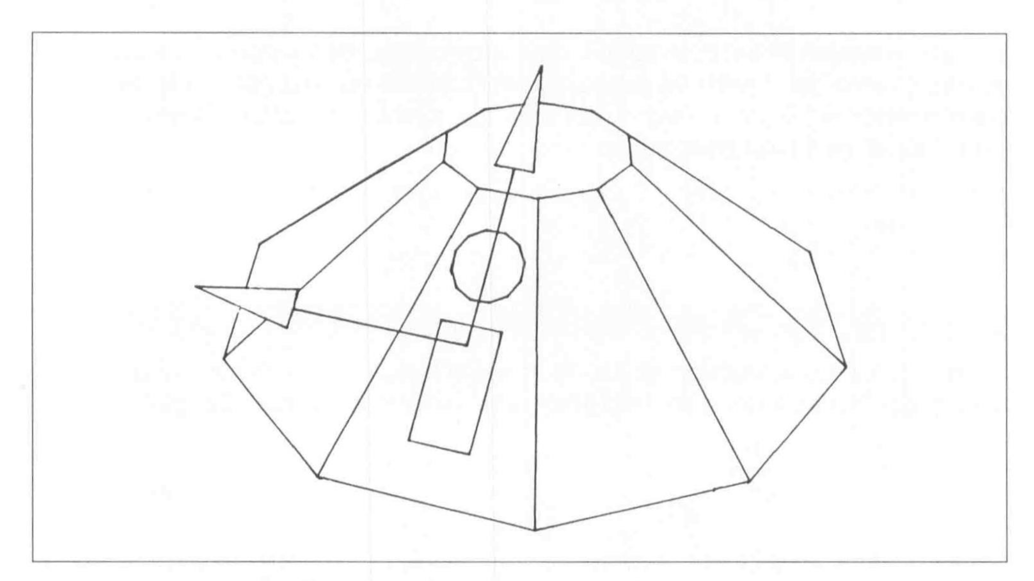


Bild 7.1: Muster auf einer Fläche

Schreiben wir daher nach der Zeile

```
PYR_STUMPF gelb drehflaeche meridian 3 0 0, 1 0 4 rot_zahl 8
```

das Schlüsselwort »muster« und dann die geheimnisvollen Zeilen:

```
muster
   flaeche 3
      polygon I_BALKEN
         gruen -0.5 -2, 0.5 -2, 0.5 2, -0.5 2
      reg. 12-Eck I_PUNKT
          gruen mitte 0 3 eckpunkt 0 3.5
```

Dazu muß natürlich etwas gesagt werden ...

Findet der 3-D-Sprinter vor den möglichen Manipulationen das Schlüsselwort »muster«, dann erwartet er sich eine Auflistung jener Flächen, auf denen etwas gezeichnet werden soll. Diese Fläche kann durch die Flächennummer festgelegt werden, in Spezialfällen auch durch das Schlüsselwort »boden« oder »deckel«. Nun folgt das Schlüsselwort »polygon« (oder, wenn es nur eine Strecke ist, das Wörtchen »strecke« bzw. »reg. 12-Eck«, wenn es ein regelmäßiges Vieleck ist). Um selbst die Übersicht nicht zu verlieren, schreibt man daraufhin den Namen des (Teil-)Musters hin, anschließend die Farbe und dann die Koordinaten der einzelnen Punkte. Doch halt, wieso auf einmal nur zweidimensionale Koordinaten?

Nun, die Geschichte ist so: Es erweist sich als überaus langwierig, die dreidimensionalen Koordinaten auszurechnen, überhaupt dann, wenn die Fläche allgemein im Raum und nicht mehr parallel zu einer Koordinatenebene liegt. Daher macht der 3-D-Sprinter folgendes (betrachten Sie dazu Bild 7.1): Es wird zuerst der

Schwerpunkt der Fläche (sagen wir grob: der Mittelpunkt) ausgerechnet. Dann wird über die Fläche ein zweidimensionales Koordinatennetz gespannt, wobei die x-Achse horizontal ist und die y-Achse »hinaufschaut«. Und auf dieses Koordinatensystem beziehen sich die zweidimensionalen Koordinaten.

Doch zurück zum »i« auf der dritten Pyramidenfläche. Der »Punkt auf dem i« soll doch ein Kreis sein. O.K., ein regelmäßiges Zwölfeck tut's auch. Aber die Koordinaten eines regelmäßigen Zwölfecks auf mindestens drei Kommastellen hinschreiben? Dann lieber nur den Mittelpunkt und einen Eckpunkt angeben (beides im »lokalen« zweidimensionalen Koordinatensystem). Das Wörtchen »reg.« (»regelmäßiges« war denn doch etwas zu lang) war das Schlüsselwort, gefolgt von der Eckenanzahl, Name und Farbe sowie den Schlüsselwörtern »mitte« und »eckpunkt«.

Was Sie unbedingt beachten sollten: Die Muster müssen (zumindest in der jetzigen Version des 3-D-Sprinters) konvex sein, sonst gibt es bei den Schlagschatten unter Umständen Fehler. Dementsprechend müssen z. B. Buchstabe L bzw. A in zwei Rechtecke bzw. sogar drei Polygone zerlegt werden. In der nächsten Version des 3-D-Sprinters wird es genügen, einfach »buchstabe A« hinzuschreiben ...

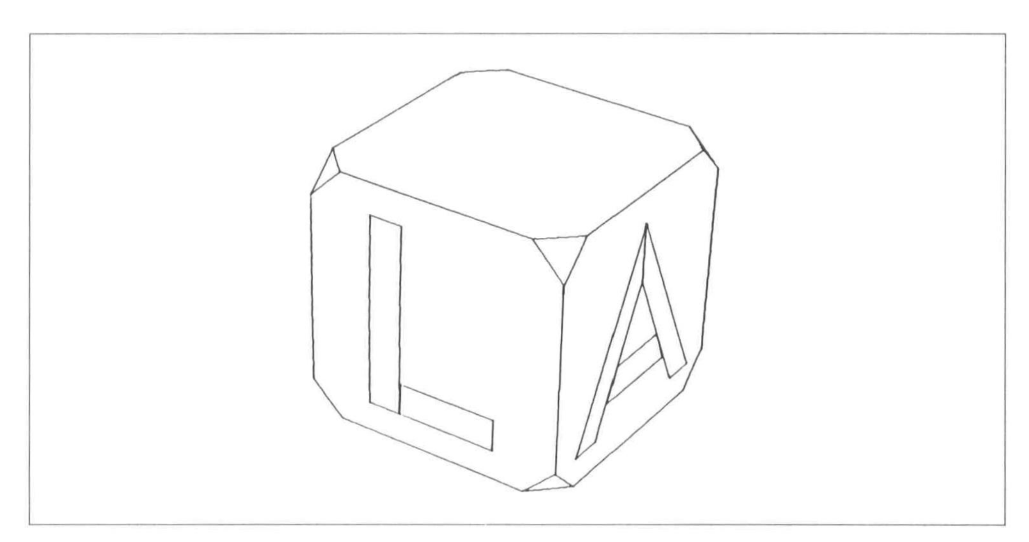


Bild 7.2: Vorläufig müssen nichtkonvexe Muster in konvexe zerlegt werden

Der große Vorteil der Muster ist, daß diese Ihre Zeichnung um einiges aufpolieren (die Pokerwürfel in Farbbild 15 machen sich z. B. recht nett ...), die Rechenzeit aber nicht besonders beeinflussen. Wenn immer es geht, sollte man daher mit aufgemalten Mustern arbeiten. Unter das Kapitel Muster wurde auch der Spezialfall

genommen, daß man einzelne Facetten einer Fläche mit einer anderen Farbe versehen will. Dies geschieht durch die Zeilen

```
muster
   flaeche 3 anmalen VORDERFLAECHE gruen
```

Also: Nach der Flächennummer das Wörtchen anmalen, der Fläche einen sinnvollen Namen geben (wegen der Leserlichkeit Ihres Files), plus die Farbe, welche die Facette haben soll. Diese Art von Muster hat dann überhaupt keinen Einfluß auf die Rechenzeit mehr.

7.4 Spiegelungen

Beim Schreiben dieses Buches sind uns jede Menge Ideen gekommen, deren Verwirklichung aus Zeitmangel einfach nur teilweise möglich war. Eine davon waren die Spiegelungen. Das klingt so einfach und erwies sich als ziemlich harter Brocken. Aber immerhin: Es funktioniert im »shade-modus« und sieht toll aus. Eine Herausforderung an die Raytracing-Leute, ihre Programme ein bißchen schneller zu machen.

Was ist zu tun? Schreiben Sie einfach in Ihr Datenfile irgendwo bei der Perspektive das Schlüsselwort »spiegel« hin, und geben Sie daraufhin den Spiegel wie ein Polygon (Kapitel 3.6.7) ein, z. B.

```
spiegel rechteck normal z bezugspunkt -4 -6 -2 seiten 8 12
```

Nur bedenken Sie, daß diese Möglichkeit des Programms noch nicht perfekt ausgereift ist. Es wird z. B. noch nicht überprüft, ob der Spiegel die Bausteine schneidet, was natürlich falsche Bilder nach sich zieht. Und dann kann es auch leicht passieren, daß Flächen zu groß werden und die Hardware des Amiga den »Guru zu Rate ziehen« muß. Beginnen Sie daher mit einem Vergrößerungsfaktor 0.5 und erhöhen Sie langsam so viel, daß die meisten Flächen noch am Bildschirm zu sehen sind. Vor allem darf der Spiegel nicht an drei Stellen über den Bildschirmrand hinausragen!

Nachdem Spiegelung in Kombination mit Schlagschatten derzeit noch nicht eingebaut ist, läßt der 3-D-Sprinter einfach den Spiegel weg, wenn Sie mit der <c>-Taste in den Schlagschatten-Modus überwechseln.

7.5 Mehr über mathematische Flächen

In Kapitel 3 wurde bereits erwähnt, daß der 3-D-Sprinter gut geeignet ist, mathematische Flächen zu zeichnen, insbesondere Drehflächen und Funktionsgraphen. Als Typenbezeichnung wird dann das Wort mathematisch verwendet.

7.5.1 Drehflächen

Sie wollen z. B. einen Torus darstellen (unter einem Torus versteht man jene Fläche, die von einem Kreis in einer Meridianebene bei Drehung um die Achse überstrichen wird). Dafür gibt es im Programm einen Typus »torus«. Um den Torus in Bild 7.3 zu beschreiben, geben Sie bitte folgende Zeilen ein:

```
MEIN_ERSTER_TORUS gelb mathematisch konkav
torus
   rad1       5    /* Radius des Mittenkreises */
   rad2       3    /* Radius des rotierenden Kreises */
   alpha1     0    /* Anfangswinkel am Mittenkreis */
   alpha2 270      /* Endwinkel am Mittenkreis */
   beta1  -90      /* Anfangswinkel am Meridiankreis */
   beta2  270      /* Endwinkel am Meridiankreis */
   par_kreise 20 /* Anzahl der Parallelkreise */
   mer_kreise 30 /* Rotationszahl */
```

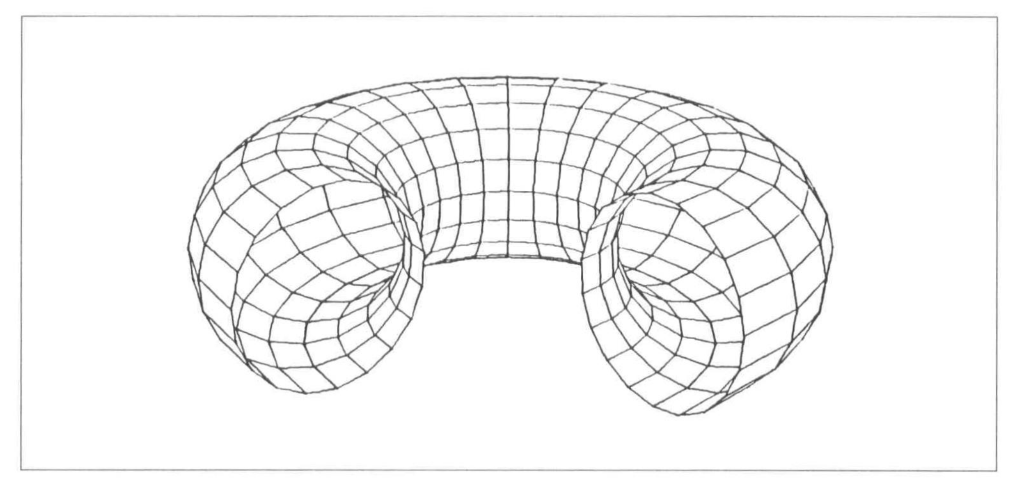


Bild 7.3: $^3/_4$-Torus

Die Proportionen des Torus können Sie verändern, wenn Sie das Verhältnis der beiden Kreisradien ändern. Wählen Sie rad2 im Verhältnis zu rad1 kleiner, dann wird der Ringdurchmesser kleiner. Wenn Sie alpha2 360 Grad wählen, ist der Torus geschlossen (Bild 7.4), mittels Variation von beta1 und beta2 können Sie bestimmen, welchen Ausschnitt des Meridiankreises Sie rotieren lassen wollen. Insbesondere liefern

```
beta1 -90  beta2 90
```

nur den Innenteil des entsprechenden Torus.

Die Wörter rad1, rad2, alpha1, alpha2, beta1, beta2, par_kreise, mer_kreise sind keine Schlüsselwörter, so daß Sie sie auch durch andere ersetzen können (nur

weggelassen dürfen sie nicht werden; dies zwingt den Anwender zum Schreiben übersichtlicher Datenzeilen). Zur Ausgabe im schattierten Modus oder auch im hidden-line-Modus gibt's noch ein paar Gags: Wenn Sie nach der Beschreibung der Fläche

```
schachbrett grau
```

dazuschreiben, legt das Programm über die Fläche ein Schachbrettmuster, bei dem die Objektfarbe und die neu dazugeschriebene Farbe verwendet werden (siehe Farbbild 18). Das Schlüsselwort »geschlitzt« bewirkt, daß der Torus längs Meridianebenen geschlitzt wird, was auch ganz nett aussehen kann (Bild 7.4b), weiterhin gibt es noch das Wort »durchloechert«, was zur Folge hat, daß jede zweite Facette weggelassen wird (ähnlich dem Schachbrett-Muster, vgl. Farbbild 26). Der Torus ist natürlich nicht die einzig interessante Drehfläche. Wir haben dem 3-D-Sprinter als Demo-Beispiele noch zwei verschiedene Arten von »Vase« eingebaut (vase1 entsteht durch Rotation einer Sinuslinie, vase2 durch Aneinandersetzen zweier Torusteile; siehe Farbbilder 26a, 26b). Bei diesen Typen brauchen Sie sich um spezielle Zusatzzahlen nicht zu kümmern. Es genügt z. B.

```
ERSTE_VASE gruen mathematisch konkav
drehflaeche vase1
   par_kreise 30
   mer_linien 25
   schachbrett gelb
   translation x -5
ZWEITE_VASE grau mathematisch konkav
drehflaeche vase2
   par_kreise 30
   mer_linien 25
   geschlitzt
   translation x 5
```

Für Sie wird es aber erst richtig interessant, wenn Sie selbst Punkte am Meridian vorgeben können. Schreiben Sie dazu statt der Typen vase1 oder vase2 einfach das Schlüsselwort »meridian« und geben Sie danach die Koordinaten einer beliebigen Anzahl von Punkten auf dem Meridian (dreidimensional) an. Wählen Sie am Anfang bitte immer die y-Koordiante 0, sonst kommt nichts Vernünftiges dabei heraus! Der 3-D-Sprinter verbindet die Punkte mittels sogenannter »Spline-Kurven« (Abschnitt 7.6), wodurch der Meridian geglättet wird. Als Beispiel diene das Weinglas in Farbbild 19 (das Listing finden Sie im Anhang!). Bild 7.4 zeigt noch ein weiteres Anwendungsbeispiel. Eine Einschränkung gilt vorläufig leider noch: Der Meridian sollte nicht verschlungen sein, oder noch besser: Wenn es nicht zu jedem z-Wert gleich viele Punkte am Meridian gibt, kann's vorläufig im shade-Modus Schwierigkeiten geben!

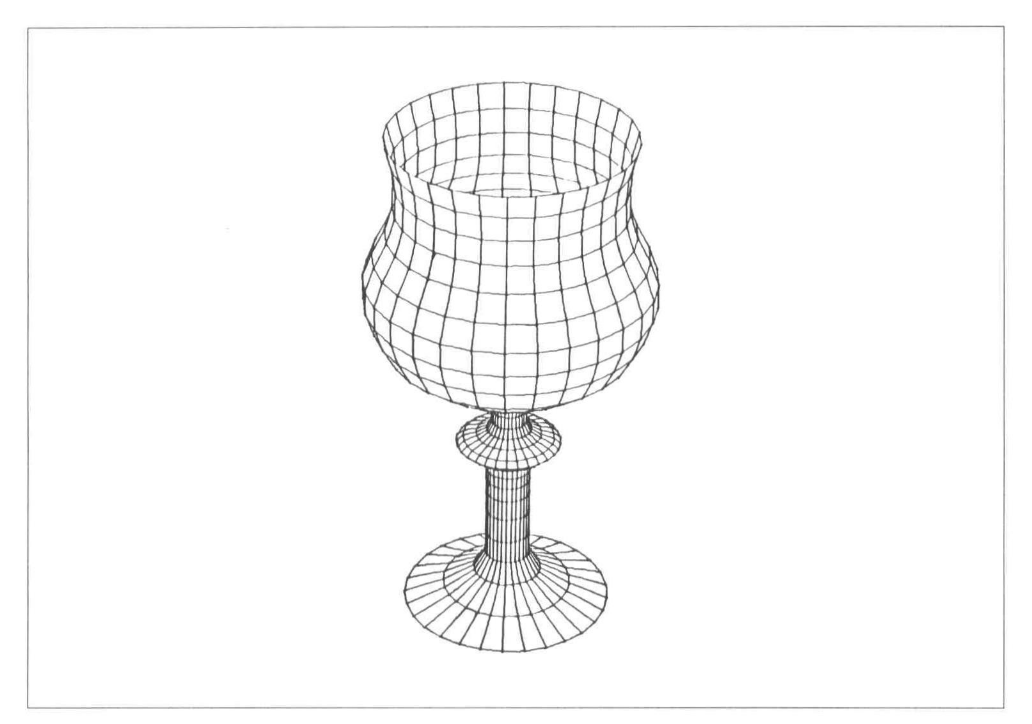


Bild 7.4: Drehfläche, von der ein gutes Dutzend Punkte am Meridian vorgegeben wurde

7.5.2 Funktionsgraphen

Eine für Techniker bzw. Wissenschaftler besonders wichtige Gattung mathematischer Flächen sind die Funktionsgraphen. Sie sind dadurch gekennzeichnet, daß es zu jedem Punkt innerhalb eines Gebiets der xy-Ebene genau einen z-Wert gibt. Im allgemeinen wählt man als Basisgebiet ein Rechteck. Beginnen wir bei der Erklärung wieder mit einem eingebauten Typ, dem Typus »berge«:

```
MEIN_ERSTER_FUNKTIONSGRAPH gelb mathematisch konkav
 funktion berge
   x_grenzen -3 3    /* x-Grenzen des Basisrechtecks */
   y_grenzen -2 2    /* y-Grenzen des Basisrechtecks */
   x_schnitte 25     /* Schnitte parallel zur x-Achse */
   y_schnitte 15     /* Schnitte parallel zur y-Achse */
```

Das Schlüsselwort »berge« bringt Computer dazu, eine gewisse mathematische Gleichung anzuspringen, die zu jedem Punkt innerhalb des Basisrechtecks einen speziellen z-Wert liefert (Farbbilder 28, 28a). Wählen Sie am Anfang die Zahl der Schnitte eher klein, damit Sie nicht allzulange auf Ihr Bild warten müssen (im shade-Modus geht's ja relativ flott, aber Vorsicht beim hidden-line-Modus: Dort nimmt die Rechenzeit quadratisch mit der Anzahl der Schnitte zu! Verwenden Sie

diesen Modus eher nur bei Ausgabe durch den Plotter). Durch wahlweises Hinzufügen der Wörter

```
schichtenlinien 20
loxodromen 10 winkel 45
            /* Linien mit festem Kurswinkel 40 Grad zu den
               Schichtenlinien */
anstieg 20 winkel 8
           /* Boeschungslinien mit 8 Grad Anstieg */
fallinien 15
```

können Sie auch spezielle Flächenkurven berechnen lassen. Am ehesten werden Sie von den Schichtenlinien Gebrauch machen, die anderen Linien sollen hier nur der Vollständigkeit halber erwähnt werden. Wer mehr darüber wissen will, sei auf die Literaturangaben in Anhang F verwiesen.

Irgendwann werden Sie sich an den »Bergen« satt gesehen haben. Dann könnten Sie noch die eingebaute Potentialfläche begutachten, die Sie mit dem Schlüsselwort »potential« (Farbbild 29) aufrufen. Eine »gedämpfte Welle« erhalten Sie schließlich mit dem Schlüsselwort »welle« (Farbbild 30). Viel interessanter ist es allerdings, selbst Funktionsgraphen zu definieren. Schreiben Sie also statt der Schlüsselwörter »berge« oder »potential« das Wort »formel« und fügen Sie dann irgendeine Formel dazu. Hier ein paar Beispiele:

```
formel sin(x)*((x*x)+(y*y)-1.5)
       /* liefert die »Spinne« in Farbbild 17 */
formel (x*x)-(y*y)
       /* Hyperbolisches Paraboloid (HP-Fläche) */
formel x*cos(x*y)
```

Verwenden Sie aber bitte eher zu viele als zu wenige Klammern. Der eingebaute Formel-Interpreter setzt nämlich nicht zwingend die Punkt- vor die Strichrechnung, das heißt

```
formel 2+x*y    /* statt 2+(x*y) */
```

wäre für den 3-D-Sprinter (2+x)*y. Ansonsten können Sie sich bei den Formeln austoben, wenn Sie daran denken, daß Divisionen durch Null vermieden werden sollten (das Programm fängt diese zwar ab, um den »Guru nicht ständig grüßen lassen« zu müssen, das Ergebnis ist dann aber meist unansehnlich).

Die letzte Möglichkeit der Eingabe der z-Werte hat am ehesten in der Praxis Bedeutung, wo statt einer Formel nur eine große Anzahl von Meßwerten zur Verfügung steht. Schreiben Sie in diesem Fall statt »formel« das Wort »datafile«, und geben Sie dann den Namen jenes Files an, in dem die z-Werte der Reihe nach anstehen. Die Werte müssen dort wie folgt angeordnet sein (ohne Satzzeichen dazwischen):

```
z(x1,    y1) z(x1,    y1+dy) ... z(x1,    y2)
z(x1+dx,y1) z(x1+dx,y1+dy) ... z(x1+dx,y2)
...
z(x2,    y1) z(x2,    y1+dy) ... z(x2,    y2)
```

x1 und y1 sind dabei die minimalen, x2 und y2 die maximalen Koordinaten des Basisrechtecks, die Zuwächse dx und dy hängen von der Anzahl n1 bzw. n2 der Schnitte parallel zur x-Achse bzw. y-Achse ab: dx = (x2-x1)/(n1-1), dy = (y2-y1)/(n2-1). Das Zahlenschema muß genau n1 * n2 Zahlen beinhalten!

Auf der mitgelieferten Datendiskette sind zwei hübsche Beispiele zu finden: Die Datenfiles »wueste.data« und »huegel.data«, die ihrerseits die Files »wueste.z« bzw. »huegel.z« aufrufen, in denen die entsprechenden z-Koordinaten aufgelistet sind (Farbbilder 31 und 32). Wie man zu solchen Datenfiles kommt? Unter uns gesagt, die Punkte wurden irgendwann mit »wüsten« mathematischen Funktionen berechnet (daher wohl der Name für das Datenfile?) und einfach abgespeichert. Sie können natürlich auch in irgendwelche Vermessungsämter gehen und Datensätze kaufen (soll es geben)!

7.6 Spline-Kurven (Interpolationskurven)

Unter Splinekurven verstehen wir ebene oder (Raum-)Kurven, die mehr oder weniger genau durch vorgegebene Punkte gehen. Das mehr oder weniger genau hängt von der Art der Anwendung ab. Wenn man z. B. von einer Meßkurve Dutzende Punkte exakt ermittelt hat, wird man durch diese Punkte eine Kurve legen, die genau durch die Meßpunkte geht. Wenn es aber darum geht, besonders glatte Kurven von gewissem Aussehen zu modellieren, bei denen es weniger entscheidend ist, daß Zwischenpunkte gewisse Koordinaten haben, wird man andere Interpolationen bevorzugen.

Der 3-D-Sprinter legt durch vorgegebene Punkte Kurven, die nicht allzusehr »ausschwingen« (für Eingeweihte: Es handelt sich um sogenannte »Akima-Splinekurven«). Dies hat für viele Zwecke Vorteile. Einer dieser Vorteile kam uns schon im Kapitel 7.6.1 bei der Erzeugung von Meridianen von Drehflächen zugute.

Sie können selbst Splinekurven erzeugen und begutachten. Als Beispiel sei ein Datenfile aufgelistet, das eine Schraublinie interpoliert, und zwar einmal mit wenigen, ein zweites Mal mit doppelt so vielen Punkten. Vergleichen Sie selbst das Ergebnis (Bild 7.5).

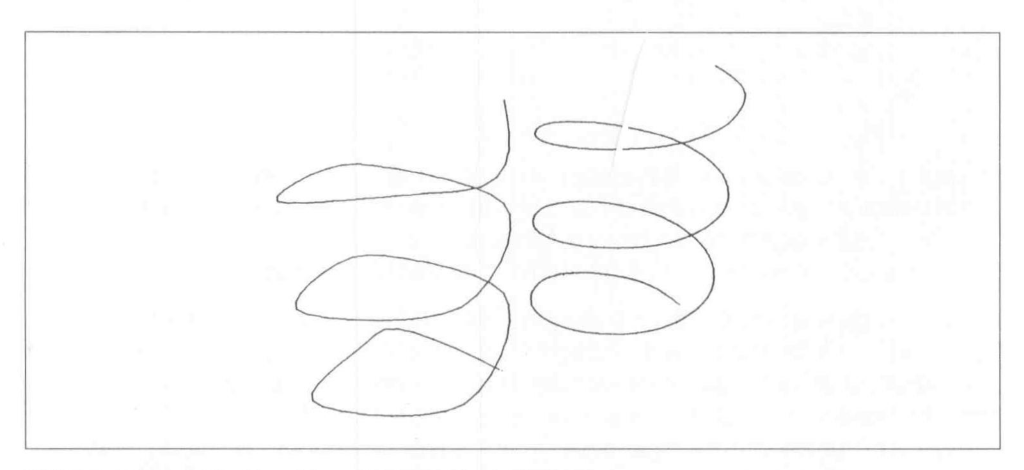


Bild 7.5: Interpolation von Raumkurven mittels Splinekurven

Das Programm braucht die Splinekurven aber intern noch viel notwendiger bei der Animation von Szenen. Darüber haben wir im Kapitel 5 schon kurz gesprochen.

7.7 Deutsch oder Englisch?

Die Version 1.2 des 3-D-Sprinters wurde in den USA auf einer IRIS-Workstation entwickelt. Was am AMIGA einigermaßen schnell aussieht, wird von so einem »Teufelsgerät« mehr oder weniger »live« am Bildschirm bewegt. Doch abgesehen davon wurde natürlich nicht nur der gesamte Quellcode in Englisch geschrieben, sondern alle Schlüsselwörter waren englisch. Sollten irgendwo im Nachrichtenfile englische Meldungen stehen, dann wissen Sie jetzt, warum. Nach ursprünglichem Zögern haben wir dem 3-D-Sprinter nun doch als Zweitsprache Deutsch beigebracht. Und wir sind froh darüber, weil die Datenfiles – und vor allem die Fehlermeldungen – einfach leichter zu lesen sind, auch wenn jeder Amiga-Freak die grundlegenden Wörter in Englisch weiß. Trotzdem hat der 3-D-Sprinter sein Englisch nicht vergessen. Im Anhang D finden Sie eine Auflistung der deutschen Schlüsselwörter mit den jeweils entsprechenden englischen Wörtern.



8 Ausplotten bzw. Ausdrucken der Zeichnungen

Wenn man sich längere Zeit dahintergeklemmt hat, um ein Datenfile zu erstellen, und dafür die optimale Ansicht und den besten Lichteinfallwinkel ausgeklügelt hat, will man natürlich dieses kleine Kunstwerk der Nachwelt nicht vorenthalten. Ein einfacher Weg dazu ist eine

8.1 Hardcopy vom Bildschirm

Halten Sie sich dazu genau an folgende Regel:

1) Richten Sie sich auf der Workbench das Piktogramm des »GraphicDump«-Programms »griffbereit« her. Auf der Programmdiskette des 3-D-Sprinters befindet es sich in der Schublade »System«.
2) Starten Sie den 3-D-Sprinter wie gewohnt und manipulieren Sie die Objekte so lange, bis sie sich in der von Ihnen gewünschten Lage befinden. Nun drücken Sie [Shift]+ [h] (wie Hardcopy) und dann keine Taste mehr!
3) Das nächste Bild erscheint nun auf weißem Hintergrund. Durch die Tastenkombination [LAmiga]+ [n] bringen Sie den Workbench-Screen in den Vordergrund. Klicken Sie mit der Maus doppelt auf das vorbereitete Piktogramm des »GraphicDump«-Programms, und wechseln Sie dann möglichst rasch (innerhalb von maximal 10 Sekunden) mit [LAmiga]+ [m] auf den Screen des 3-D-Sprinters zurück. Klicken Sie zur Sicherheit zusätzlich einmal mit der linken Maus-Taste in den Screen.
4) Warten Sie, bis der Drucker seinen Job getan hat, und drücken Sie anschließend eine beliebige Taste, um in die Animation zurückzukehren.

Die Hardcopy ist für alle Zeichenmodi, die der 3-D-Sprinter anbietet, geeignet. Also insbesondere auch für schattierte Bilder.

8.2 Ausgabe über den Plotter

Sollten Sie glücklicher Besitzer eines Plotters sein, und haben Sie diesen nach Anleitung in Kapitel 4.6.2 installiert, dann wird die Sache noch einfacher (allerdings eingeschränkt auf den hidden-line-Modus bzw. natürlich auch das Drahtgittermodell):

1) Drücken Sie die [h]- oder [w]-Taste, um eine Liniengrafik zu erhalten.
2) Drücken Sie die [p]-Taste, um den Plotter zu aktivieren.

Sollte aus irgendwelchen Gründen der Plotter nicht sofort ansprechen, versuchen Sie's beim nächsten Bild nochmal mit der [p]-Taste.

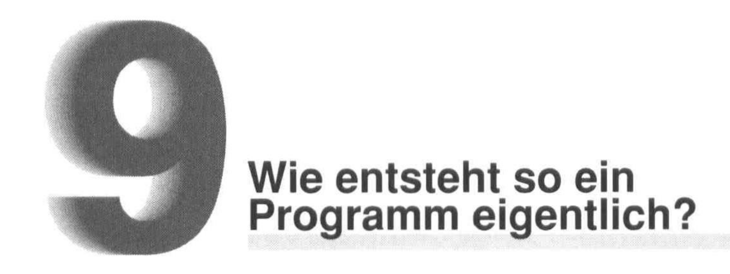
8.3 Abfotografieren vom Bildschirm

Eine einfache Methode, farbige Bilder naturgetreu zu kopieren, besteht darin, den Bildschirm einfach abzufotografieren. So entstanden die Abbildungen im Anhang.

Beachten Sie dabei folgende Punkte:

1) Der Raum muß vollständig abgedunkelt sein (am besten eignen sich die späten Abendstunden).
2) Arbeiten Sie auf jeden Fall mit Stativ (die Belichtungszeiten liegen um eine halbe Sekunde). Justieren Sie die Kamera genau senkrecht zum Bildschirm (wichtig!)
3) Die Belichtungszeiten liegen je nach Helligkeit des Bildes bei einem 100-ASA-Film (21 DIN) zwischen einer Achtel- und einer halben Sekunde. Bei einer Automatikkamera stellen Sie am besten auf Automatik.

Dias und Farbbilder sollten dann gleichermaßen gut gelingen.



9 Wie entsteht so ein Programm eigentlich?

Nach der grauen Theorie und bunten Praxis jetzt eine etwas heitere Geschichte zum Abschluß.

9.1 Es war einmal vor langer Zeit ... C64 und Basic

Es war einmal vor langer Zeit ...

Halt, Stop, Cancel, Reset: Wir wollen hier kein Märchen erzählen, sondern von der »harten« Wirklichkeit berichten. Begonnen hat unser Leidensweg am Computer vor vier Jahren mit einem C64. Auf diesem wurden in Simons Basic die ersten Algorithmen zur Darstellung von dreidimensionalen Objekten entwickelt. Allerdings war damals von Animation noch keine Rede, wir waren froh, wenn ein Bild in weniger als zwei Minuten fertig war. Manchmal wurde aber auch eine ganze Nacht gerechnet. Diese Algorithmen wurden in einem Buch zusammengefaßt (siehe Anhang: weiterführende Literatur). Dieses Buch und andere Veröffentlichungen ermöglichten Georg ein Forschungsstipendium für Computergrafik an der Universität Princeton New Jersey in den USA.

9.2 Amerika, Workstations und C

Die Großrechenanlagen (Workstations) an der Universität Princeton werden praktisch nur in C programmiert. Diese Programmiersprache faszinierte Georg sofort, da sie es erlaubt, mit einer gewissen Disziplin völlig systemunabhängige Programme (wie es zum Beispiel der 3-D-Sprinter ist) zu schreiben.

Nach dem Übertragen der am C64 entwickelten Algorithmen in die Programmiersprache C, begann Georg mit der Entwicklung der Schattenroutinen. Zuerst wurde ein Programm geschrieben, das Pseudo-Schatten von konvexen Körpern auf eine ebene Fläche warf. Dieses wurde anschließend so weiterentwickelt, daß es die Schatten, die beliebige Körper auf andere warfen, berechnen und zeichnen konnte.

Das Programm wurde auf einer »Iris«-Workstation geschrieben. Um seine Systemunabhängigkeit zu testen, wurde es von Zeit zu Zeit auf eine »Sun«-Workstation übertragen. Nachdem es korrekt arbeitete, begann Georg mit Hilfe von Prof. Silvio Levy vom Mathematischen Institut der Universität Princeton mit der

Optimierung des Programms. Berechnungen, bisher zwei- oder mehrmals gemacht, wurden nun mehr einmal durchgeführt. In den Algorithmus zum Sortieren der Objekte wurden neue Optionen eingebaut u.v.m.

In der Zwischenzeit hatte Thomas unabhängig davon mit der Übertragung der Algorithmen vom C64 auf den Amiga (in Amiga Basic) begonnen. Dies wurde aber verworfen, als Georg mit den verbesserten Versionen aus den USA zurückkam.

9.3 Ein langer Nachmittag – »TRY« entstand auf dem Amiga

Als das Programm auf der Workstation (Impuls Workstation und Micro Vax) der Abteilung für Computergrafik an der Österreichischen Akademie der Wissenschaften lief, wurde mit der Umsetzung auf den Amiga begonnen. Nachdem einige Routinen wegen Speichermangels umgeschrieben werden mußten und das Programm dann einwandfrei lief, trauten wir unseren Augen nicht mehr. Der Amiga war »nur« ca. zehnmal so langsam wie die Workstation!

Insgesamt benötigte die Umsetzung des Grundprogrammgrundstocks nur wenige Tage. Dies war möglich, weil das Programm von Anfang an nur im systemunabhängigen C-Code geschrieben wurde.

Nach dieser Überraschung machte die Arbeit gleich doppelt so viel Spaß, wenn sie auch sehr mühsam war, da wir vom Amiga ungefähr so viel Ahnung hatten, wie eine Schnecke vom Hürdenlauf.

Dennoch schafften wir es bis zum Vorabend einer ersten Präsentation anläßlich eines »Tages der offenen Tür« an der Schule, an der wir beschäftigt sind, den Animationseffekt durch »Double Buffering« bzw. »Page-Flipping« zu erreichen.

Das ständige Herumprobieren mit Datenfiles veranlaßte uns an diesem Tag, das Programm »TRY« (engl.: versuchen, probieren) zu nennen.

9.4 Vom Cli zur Workbench oder Intuition wird aktiv

Was wäre ein Programm am Amiga, wenn es nicht die grafische Benutzeroberfläche Intuition verwenden würde. Da es aber weiterhin systemunabhängig bleiben sollte, wurde einfach ein zweites Programm geschrieben: Die grafische Benutzeroberfläche. Diese ruft einfach das systemunabhängige Programm »TRY« mit den richtigen Optionen auf, nachdem Sie alle Einstellungen getätigt haben.

Als das Programm einigermaßen lief, sprachen wir bei Commodore Österreich vor (bzw. unser Programm zeichnete vor), um in den Kreis der Amiga-Entwickler aufgenommen zu werden.

Nach einigen Stunden Vorführarbeit waren wir »Registered Amiga Developpers« (so die offizielle Bezeichnung; man könnte uns auch simpel als Amiga-Freaks bezeichnen). Dies eröffnete uns völlig neue Möglichkeiten, wir konnten unser Programm auf den verschiedensten Amigas (z. B. Amiga 2500) ausprobieren und es so umschreiben, daß es auch mit zukünftigen Versionen des Amiga-Betriebssystems laufen wird.

Wichtig war auch, daß wir uns mit anderen Amiga-Entwicklern treffen und Erfahrungen austauschen konnten. In der Zwischenzeit war uns der Amiga so ans Herz gewachsen, daß wir nicht mehr auf der Workstation das Programm weiterentwickelten, sondern auf dem Amiga.

9.5 Die Testphase

Diese bescherte uns eine Menge der netten rot blinkenden Meldungen:

```
Guru Meditation          press left mouse button to continue
             82011234                 0000ffff
```

Obwohl Meditationszeiten recht interessant sind (es ist wirklich erstaunlich, auf wie viele verschiedene Arten der Amiga abstürzen kann), kosteten sie uns manchmal zu viele Nerven. Insbesonders bei der Entwicklung der Routinen zur Spiegelung mußten wir das Programm wieder auf der Workstation weiterentwickeln, nachdem der Amiga zwanzigmal oder öfter in Serie abstürzte und es unmöglich war, die Fehlerursache festzustellen. Georg murmelte in seiner Verzweiflung sogar schon vom Verkauf seines Amigas.

Deshalb entschlossen wir uns, doch wieder auf der Workstation weiterzuarbeiten. Sobald wieder eine lauffähige Version vorlag, wurde sie auf den Amiga überspielt und wieder auf ihm weiterentwickelt (es geht doch nichts über zwei oder mehr Fenster, in denen gleichzeitig gearbeitet werden kann).

Nachdem mit diesen Zeilen des Buches die Arbeit fast beendet ist (nun muß alles noch ausgedruckt und dem Verlag zugeschickt werden), wollen wir uns von Ihnen für den Augenblick verabschieden, denn in unseren Köpfen sammeln sich schon wieder neue Ideen, die nur darauf warten, in Bits und Bytes verwandelt zu werden.

Tschüs und auf bald

Georg & Thomas

PS: Wir können es uns ja doch nicht verkneifen. Hier einige unserer neuen Ideen:

9.6 Immer neue Ideen

- Verbesserung der Spiegelungen: Eine oder mehrere Flächen eines Objektes werden als Spiegel erklärt, und die Schatten werden auch im Spiegelbild zu sehen sein.
- Anaglyphen: Das sind 3-D-Grafiken, die, wenn sie mit einer Rot-Grün-Brille betrachtet werden, wirklich im Raum stehen. Auch eine Lösung in Farbe steht zur Diskussion!
- Neue Objekttypen: Einrichtungsgegenstände (Möbel, Tisch, Topfpflanzen), neue mathematische Flächen (Schraubflächen).
- Mehr als eine Lichtquelle
- Ein Objekt-Editor, der die Konstruktion der Objekte mit der Maus ermöglicht.
- u.v.m ...



Anhang

Anhang A

Jede Menge Beispiel

Listing zu Farbbild 1:

```
ATOMIUM_auf_einem_Spiegel

punkte 3000
augpunkt 40 25 25
licht 300 -200 400
spiegel quadrat normal z bezugspunkt -14 -14 -12 seite 28
vergroesserung 1.4
fixiere 0 0 -3

26 bausteine

KUGEL1 gruen kugel radius 3.0 meridian_groesse 10 rot_zahl 15
gruppe OBEN {
   KUGEL2 gelb wie KUGEL1 translation z 9
   ZYL1 grau drehflaeche voll meridian 0.6 0 2.95, 0.6 0 6.05 rot_zahl 10
}
gruppe UNTEN wie OBEN rotation x 180
gruppe VORNE wie OBEN rotation y  90
gruppe HINTEN wie OBEN rotation y -90
gruppe RECHTS wie OBEN rotation x 90
gruppe LINKS wie OBEN rotation x -90
gruppe ZYL_OBEN {
   ZYL2 grau drehflaeche voll meridian 0.6 0 -0.85, 0.6 0 6.05 rot_zahl 10
      rotation achsenpunkt 0 0 9 achsenrichtung y winkel 45
   ZYL3 grau wie ZYL2 rotation z  90
   ZYL4 grau wie ZYL2 rotation z 180
   ZYL5 grau wie ZYL2 rotation z 270
}
gruppe ZYL_UNTEN wie ZYL_OBEN rotation x 180

gruppe HORIZONTALE_ZYL {
   ZYL6 grau wie ZYL2 rotation x 90
   ZYL7 grau wie ZYL6 rotation z 90
   ZYL8 grau wie ZYL6 rotation z 180
   ZYL9 grau wie ZYL6 rotation z 270
}
/*
/* Spiegel entfernen und diese Kommentare entfernen stellt das Atomium
   auf eine solide Basisebene */
BASISEBENE gruen /*quadrat normal z bezugspunkt -14 -14 -12 seite 28*/
   drehflaeche voll meridian -14 -14 -14, -14 -14 -12 rot_zahl 4
*/
```

Listing zu Farbbild 2:

```
PYRAMIDEN_BLUETE_auf_einem_Spiegel

augpunkt -25 -10 15
licht -400 150 300
vergroesserung 1.1
spiegel quadrat normal z bezugspunkt -8 -8 -8 seite 16
fixiere 0 0 -4

12 bausteine

gruppe OBERE_PYRAMIDEN {
   1.pyramide gelb drehflaeche voll meridian 0 2 4, 0 1 7 rot_zahl 5
   2.pyramide gruen wie 1.pyramide  rotation  x -63.4
   3.pyramide gelb wie 2.pyramide  rotation   z 72
   4.pyramide grau wie 3.pyramide rotation    z 72
   5.pyramide weiss wie 4.pyramide  rotation  z 72
   6.pyramide schwarz wie 5.pyramide rotation z 72
}
gruppe UNTERE_PYRAMIDEN wie OBERE_PYRAMIDEN rotation x 180
```

Listing zu Farbbild 3:

```
BOHRFUTTER_MIT_SCHRAUBENSCHLUESSEL

augpunkt 20 20 15
licht 100 -100 200
spiegel rechteck normal y bezugspunkt 10 -4 -6 seiten 20 12
vergroesserung 1.4
fixiere 0 -4 0.5

gruppe UNTERTEIL {
   KEGEL1 grau drehflaeche voll meridian 1 0 -6, 1.8 0 0 rot_zahl 16
   ZYL grau drehflaeche voll meridian 0.8 0 -6.3, 0.8 0 -6 rot_zahl 16
   KEGEL2 grau drehflaeche voll meridian 1.0 0 -7.3, 0.9 0 -6.3 rot_zahl 16
}

gruppe MITTELSCHEIBE {
   TEIL1 gruen translation voll
      basis 0.5 -2.45 0, 1.25 -2.165 0, 1.915 -1.605 0, 2.35 -0.85 0,
                 2.35 0.85 0, 1.915 1.605 0, 1.25 2.165 0, 0.5 2.45 0
      schub_vektor z 0.5
   TEIL2 gruen wie TEIL1 rotation y 180  translation z 0.5
   MITTELTEIL gruen box voll 1 4 0.5 translation -0.5 -2 0
} rotation z -60

gruppe OBERTEIL {
   ZYL1 grau drehflaeche voll meridian 1.7 0 0.5, 1.7 0 1 rot_zahl 16
   ZYL2 grau drehflaeche voll meridian 1.2 0 1, 1.2 0 2 rot_zahl 16
   ZYL_KEGEL1 grau drehflaeche voll
       meridian 1.2 0 2, 1.4 0 2.2, 1.4 0 3 rot_zahl 16
   ZYL_KEGEL2 grau drehflaeche voll
       meridian 1.4 0 3, 1.7 0 3.3, 1.7 0 5.3, 1.2 0 6.3 , 1 0 6.3 rot_zahl 16
   /* Pseudoschattierung auf der obersten Flaeche */
```

```
   muster flaeche 66 strecke deckel schwarz -1 0 , 0 0
                     strecke deckel schwarz 0.5 0.87 , 0 0
                     strecke deckel schwarz 0.5 -0.87 , 0 0
                     anmalen deckel gruen
} rotation z -60
gruppe SCHRAUBEN_SCHLUESSEL {
   BACKE1 gelb translation voll
      basis 2.35 -1.5 0, 3.35 -0.8 0, 3.35 2.5 0, 2.35 1.4 0
   schub_vektor z 0.75 translation z -0.4
   BACKE2 gelb wie BACKE1 rotation y 180
   BACKENHALTER1 gelb translation voll
      basis 2.35 1.4 0, 3.35 2.5 0, 1.5 4.2 0, 0 2.8 0
   schub_vektor z 0.8 translation z -0.4
   BACKENHALTER2 gelb wie BACKENHALTER1  rotation y 180
   HEBEL gelb translation voll
      basis 0 2.8 0, 1.5 4.2 0, 1.5 13 0, -1.5 13 0, -1.5 4.2 0
   schub_vektor z 0.8 translation z -0.4
}  translation 0 0.3 0.2 rotation z -60
```

Listing zu Farbbild 4:

```
BUCHSTABEN

augpunkt 20 -10 20
licht 50 50 100

10 bausteine

gruppe BUCHSTABE_Y1 {
   Y1/1 gelb translation voll
        basis  0 -1 -2, 0 1 -2, 0 1 0,  0 0 1,  0 -1 0
        schub_vektor 3 0 0
   Y1/2 gelb translation voll
        basis  0 1 0,  0 3 2, 0 2 3,  0 0 1
        schub_vektor 3 0 0
   Y1/3 gelb wie Y1/2 rotation z 180 translation x 3
} translation -1.5 0 0

gruppe BUCHSTABE_Y2 wie BUCHSTABE_Y1
   streckung zentrum 0 0 -2 faktor 0.4
   rotation z 90  translation 3 -3 0

gruppe BUCHSTABE_Y3 wie BUCHSTABE_Y1
   streckung zentrum 0 0 -2 faktor 0.6
   rotation z 45
   translation 3 3.5 0

gruppe BASISEBENE {
   EBENE
     grau quadrat normal z bezugspunkt -5 -5 0 seite 10
     muster flaeche 1 polygon QUADRAT gruen -4 -4, -4 -2, -2 -2, -2 -4
                      polygon QUADRAT gruen  2 2, 2 4, 4 4, 4 2
                      polygon QUADRAT gruen  2 -4, 4 -4, 4 -2, 2 -2
                      polygon QUADRAT gruen  -4 2, -2 2, -2 4, -4 4
} translation  z -2
```

Listing zu Farbbild 5:

```
FLUGZEUG

augpunkt 15 -25 20
licht -100 100 100
vergroesserung 1.6

4 bausteine

gruppe FLIEGER {
   RUMPF
      grau drehflaeche voll
      meridian 0 0 -1 ,  0 1.5 6 ,  0 1.5 14 ,  0 1 16  rot_zahl 16
      translation z -8
      rotation x 90

   LINKER_FLUEGEL
      grau translation voll
      basis 1.5 0 0 ,  11.5 -4 0 , 11.5 -3 0 , 6.5 1.5 0 , 1.5 5 0
      schub_vektor z 0.2
      muster flaeche deckel reg. 6-Eck KREIS schwarz mitte 0 0 eckpunkt 0.5 0
             flaeche boden  reg. 6-Eck KREIS schwarz mitte 0 0 eckpunkt 0.5 0
      rotation z 180
      translation y 4

   RECHTER_FLUEGEL
      grau wie LINKER_FLUEGEL rotation y 180

   HECK_FLUEGEL
      grau polygon  normal x punkte 0 0 , 4 0 , 5 4 , 3.5 4
      muster flaeche 1 polygon FLAGGE gelb -0.8 0 , 1 0, 1 0.5 , -0.8 0.5
                       polygon FLAGGE gruen -0.8 -0.5 , 1 -0.5, 1 0 , -0.8 0
                       polygon FLAGGE gelb -0.8 -1 , 1 -1, 1 -0.5 , -0.8 -0.5
      translation 0 2 1.5
}
```

Listing zu Farbbild 6:

```
GESTAENGE

augpunkt 10 20 25
licht 140 -35 250
vergroesserung 1.3
fixiere zentrum

62 bausteine

gruppe BASIS {
   BODEN gruen quadrat normal z bezugspunkt -10 -10 -5 seite 20
}
```

```
gruppe 1.HALBRING {
   TEIL_1 gelb translation voll
   basis 5.2 0 0, 5.023 1.346 0, 2.801 0.7506 0, 2.9 0 0
   schub_vektor z 0.6
   TEIL_2 gelb wie TEIL_1 rotation z -15
   TEIL_3 gelb wie TEIL_1 rotation z -30
   TEIL_4 gelb wie TEIL_1 rotation z -45
   TEIL_5 gelb wie TEIL_1 rotation z -60
   TEIL_6 gelb wie TEIL_1 rotation z -75
   TEIL_7 gelb wie TEIL_1 rotation z -90
   TEIL_8 gelb wie TEIL_1 rotation z -105
   TEIL_9 gelb wie TEIL_1 rotation z -120
   TEIL_10 gelb wie TEIL_1 rotation z -135
   TEIL_11 gelb wie TEIL_1 rotation z -150
   TEIL_12 gelb wie TEIL_1 rotation z -165
}

gruppe 2.HALBRING wie 1.HALBRING rotation z 180

gruppe UNTERE_SAEULEN {
   TEIL_SAEULE_1 grau drehflaeche voll meridian 0.5 0 0, 0.5 0 5 rot_zahl 8
      translation x 4
   TEIL_SAEULE_2 grau wie TEIL_SAEULE_1 rotation z 90
   TEIL_SAEULE_2 grau wie TEIL_SAEULE_1 rotation z 180
   TEIL_SAEULE_2 grau wie TEIL_SAEULE_1 rotation z 270
} translation z -5

gruppe MITTLERE_SAEULEN {
   M1 grau drehflaeche voll meridian 0.5 0 0, 0.5 0 10 rot_zahl 8
      translation x 4
   M2 grau wie M1 rotation z 90
   M2 grau wie M1 rotation z 180
   M2 grau wie M1 rotation z 270
} translation z 0.6

gruppe OBERE_SAEULEN wie UNTERE_SAEULEN translation z 16.2

gruppe 3.HALBRING wie 1.HALBRING translation z 10.6
gruppe 4.HALBRING wie 2.HALBRING translation z 10.6
```

Listing zu Farbbild 7:

```
REIHENHAEUSER

augpunkt 30 70 40
licht 1500 500 1500
fixiere zentrum
vergroesserung 1.5

25 bausteine

gruppe UMGEBUNG {
   RASENFLAECHE
      gruen polygon normal z bezugspunkt 0 0 0
  punkte 18 -30, 18 22, -14 22, -14 -30
}
```

```
gruppe HAUS1 {
   GRUNDKOERPER
      gelb translation voll
      basis  0 0 0, 0 15 0, 0 15 11.3, 0 7.5 19.7, 0 0 11.3
      schub_vektor 22 0 0
  muster
     flaeche 7 polygon BALKONTUER1 grau -3.5 -3, -2 -3, -2 1, -3.5 1
               polygon BALKONTUER2 grau 2 -3, 3.5 -3, 3.5 1, 2 1
               polygon GARTENTUER1 grau -3.8 -8, -1.7 -8, -1.7 -4, -3.8 -4
               polygon GARTENTUER2 grau 1.7 -8, 3.8 -8, 3.8 -4, 1.7 -4
               polygon TUER grau -0.75 1.8, 0.75 1.8, 0.75 5.8, -0.75 5.8
      translation x -11
   VORSPRUNG_OST
      gelb translation voll
      basis 0 0 0, 0 0.8 0, 0 0.8 10.3, 0 0 11.2
      schub_vektor x 7.4
      translation -3.7 15 0
   DACHVORSPRUNG_OST
      gelb allgemein voll
      punkte 3.15 0 0, -3.15 0 0,  0 0 5.8, 0 -5.1 5.8
      kanten 0 2, 1 2, 0 3, 1 3, 2 3
      flaechen 0 1 2 , 1 0 3 , 0 2 3 , 1 3 2
      translation 0 15.8 10.5
   GROSSE_DACHFLAECHE_OST1
      gruen allgemein hohl
      punkte 12 0 0, 12 13.3 0, 3.7 13.3 0,  0 4.5 0, 0 0 0
      kanten 0 1, 1 2, 2 3, 4 0
      flaechen 0 1 2 3 4
      rotation x 48.5
      translation 0 7.5 19.8
   GROSSE_DACHFLAECHE_OST2
      gruen allgemein hohl
      punkte -12 0 0, -12 13.3 0, -3.75 13.3 0,  0 4.5 0, 0 0 0
      kanten 0 1, 1 2, 2 3, 4 0
      flaechen 0 1 2 3 4
      rotation x 48.5
      translation 0 7.5 19.8
   KLEINES_DACH_SUEDOST
      gruen allgemein hohl
      punkte 3.55 16.3 9.9, 0 16.3 16.5, 0 10.5 16.5
      kanten 0 1, 1 2 , 0 2
      flaechen 0 1 2
   KLEINES_DACH_NORDOST
      gruen allgemein hohl
      punkte -3.55 16.3 9.9, 0 16.3 16.5, 0 10.5 16.5
      kanten 0 1, 1 2 , 0 2
      flaechen 0 1 2
   DACHFLAECHE_WEST
      gruen allgemein hohl
      punkte 12 0 0, 12 13 0, -12 13 0, -12 0 0
      kanten 0 1, 1 2, 2 3, 3 0
      flaechen 0 1 2 3
      rotation x 132
      translation 0 7.5 19.8
   BALKON_VORDERFLAECHE
      gruen rechteck normal x bezugspunkt 13.6 2.2 5.5 seiten 10.7 2
   BALKON_SEITE1
      gruen rechteck normal y bezugspunkt 13.6 12.9 5.5 seiten 2.6 2
```

```
   BALKON_SEITE2 gruen wie BALKON_SEITE1 translation y -10.7
   BALKON_BODEN gruen rechteck normal z bezugspunkt 11 2.2 5.5 seiten 2.6 10.2
}
group HAUS2 wie HAUS1 translation y -23
```

Listing zu Farbbild 8:

```
HUBSCHRAUBER

augpunkt 17 -6 16
licht -320 -100 300
vergroesserung 1.2
fixiere zentrum

30 bausteine

gruppe KUFEN {
   KUFE1 gelb rechteck normal z bezugspunkt 1.3 -4 -1.0 seiten 0.25 7
   KUFE2 gelb wie KUFE1 translation x -2.85
   SCHAUFEL1 gelb rechteck normal z bezugspunkt 1.3 -4.5 -1 seiten 0.25 0.5
      rotation achsenpunkt 1.5 -4 -1 achsenrichtung x winkel 35
   SCHAUFEL2 gelb wie SCHAUFEL1 translation x -2.85
   STUETZE1 grau zylinder voll
      radius 0.125 left_z -0.3 0.9 right_z -0.15 1.0 rot_zahl 6
      rotation z 90
      rotation y 30
   translation -1.3 0.8 -0.8
   STUETZE2 grau wie STUETZE1 translation y -3
   STUETZE3 grau wie STUETZE1
   translation 1.3 -0.8 0.8 rotation z 180 translation 1.3 0.8 -0.8
   STUETZE4 grau wie STUETZE3 translation y -3
}

gruppe MITTELTEIL {
    SCHNAUZE grau drehflaeche voll
 drehwinkel 180 meridian 1 0 0, 1 0 1.2 rot_zahl 6
 rotation z 180 translation y -4.2
   BODY1 grau allgemein voll
      punkte 1 0 0, 1 4.2 0, -1 0 0 , -1 4.2 0, 1 4.2 1.2, -1 4.2 1.2,
          0.8 3.3 2.1, -0.8 3.3 2.1, 0.6 2.8 2.62, -0.6 2.8 2.62,
    0.25 2.5 2.9, -0.25 2.5 2.9, 1 0 2.9, -1 0 2.9
      kanten 0 1, 1 3, 3 2, 2 0, 1 4, 4 5, 5 3, 4 6, 6 8, 8 10,
          10 11, 11 9, 9 7, 7 5, 4 12, 6 12, 8 12, 10 12,
    11 13, 9 13, 7 13, 5 13, 0 12, 13 12, 2 13
      flaechen 0 1 3 2 , 0 1 4 12 , 4 2 13 5 , 10 11 13 12 ,
         4 5 7 9 11 10 8 6 , 7 5 13 , 9 7 13 , 11 9 13 ,
   10 8 12 , 8 6 12 , 6 4 12 , 0 2 13 12 , 1 3 5 4 ,
   3 5 13 2
       rotation z 180
    BODY2 grau allgemein voll
    punkte 0 0 0, 2 -0.7 0.8, 2 -0.7 2.2, 1 -0.35 2.9, 0 0 2.9,
           0 -2 2.9, 1 -1.65 2.9, 2 -1.3 2.2, 2 -1.3 0.8, 0 -2 0
       kanten
       /* A B, B C, C D, D E, E A, F A, E O, P Q, P O,
       C Q, D P, O F, F R , Q R, B R */
        0 1, 1 2, 2 3, 3 4, 4 0, 9 0, 4 5, 6 7, 6 5,
       2 7, 3 6, 5 9, 9 8 , 7 8, 1 8
```

```
          flaechen
       /* A B C D E , F R Q P O , A E O F
          R B C Q , C Q P D , D P O E , A F R B */
       0 1 2 3 4 , 9 8 7 6 5 , 0 4 5 9 ,
          8 1 2 7 , 2 7 6 3 , 3 6 5 4 , 0 9 8 1
         rotation z -90  translation x -1
} translation y 1

gruppe HINTERTEIL {
   TRAPEZ gelb translation voll
      basis 0 0 0, 0 4 0, -0.5 4 0, -1.3 0 0 schub_vektor z 0.5
      rotation y -90 translation 0.25 3 2
   HECKFLUEGEL gelb translation voll
      basis 0 9 0, 1.5 9 0, 1.5 8 0, 0.5 7 0, 0 7 0 schub_vektor z 0.5
   translation z -0.25
   rotation y -90
   translation z 1.5
}  translation z 0.15

gruppe BASIS {
   BODEN gruen quadrat normal z bezugspunkt -7 -7 0 seite 14
      /* quader voll 14 14 0.2
   translation -7 -7 0 */
} translation z -2

gruppe ROTORBLAETTER {
    BLATT1 grau rechteck normal z bezugspunkt 0 0 0 seiten 6.5 0.3
    /* quader voll 6.5 0.3 0.15 */ translation 0.5 -0.15 0
 BLATT2 grau wie BLATT1 rotation z 90
 BLATT3 grau wie BLATT1 rotation z 180
 BLATT4 grau wie BLATT1 rotation z 270
 ZYL1 gruen drehflaeche voll
    meridian 0.4 0 0, 0.4 0 0.3  rot_zahl 8
    translation z -0.15
 ZYL2 gruen drehflaeche voll
    meridian 0.2 0 -0.4, 0.2 0 0  rot_zahl 8
    translation z -0.15
} translation z 3.5 rotation z 25

gruppe STABILISATOR wie ROTORBLAETTER
  translation z -3.5 streckung zentrum 0 0 0 faktor 0.2
  rotation y 90 translation -0.5 8.3 2.4
```

Listing zu Farbbild 9:

```
ESPRESSO_MASCHINE

augpunkt -25 13 25
licht 125 105 180
vergroesserung 1.65
spiegel quadrat normal z bezugspunkt -10 -10 -11 seite 20
fixiere 0 0 -3

8 bausteine
```

```
gruppe MITTELTEIL {
   RING gelb drehflaeche hohl
      meridian 3.5 0 -3 , 5.6 0 5  rot_zahl 8
      rotation z 22.5

   PYRAMIDE grau drehflaeche voll
      meridian  5.6 0 -11 , 3.5 0 -3.01  rot_zahl 8
      rotation z 22.5
}
gruppe DECKEL {
   KNOPF grau drehflaeche voll
      meridian 0.925 0.385 6.5 , 1.2 0.5 8.5  rot_zahl 8
   DECKEL grau drehflaeche hohl
      meridian 5.15 2.2 5 , 0.925 0.385 6.5 rot_zahl 8
} rotation achsenpunkt 0 5.15 5.05 achsenrichtung x angle 45

gruppe HENKEL {
   ANSATZ grau zylinder voll
      radius 0.7  z_links 0 3.4   z_rechts  0 2 rot_zahl 6
      rotation x 90 translation y -2.7
   MITTELSTUECK grau zylinder voll
      radius 0.7  z_links 0 5   z_rechts  -1.4 6.4 rot_zahl 6
      translation z -5.7
   ENDSTUECK grau zylinder voll
      radius 0.7  z_links 0 1   z_rechts  0 2.4 rot_zahl 6
      rotation x 90  translation 0 -1.7 -6.4
} rotation x 14 translation 0 7.5 3.3
```

Listing zu Farbbild 10:

```
DORFKIRCHE

augpunkt 20 17 13
licht -50 170 230
fixiere 3 0 4

20 bausteine

gruppe TURMTRAKT {
   LinkerTurm gelb drehflaeche voll
      meridian 1 1 0 , 1 1 10, 0 0 12.5    rot_zahl 4
  muster
     flaeche 1 polygon FENSTER gruen  0.1 3.2,  0.6 3.2,  0.6 4.1,  0.1 4.1
            polygon FENSTER gruen -0.6 3.2, -0.1 3.2, -0.1 4.1, -0.6 4.1
     flaeche 2 polygon FENSTER gruen  0.1 3.2,  0.6 3.2,  0.6 4.1,  0.1 4.1
            polygon FENSTER gruen -0.6 3.2, -0.1 3.2, -0.1 4.1, -0.6 4.1
     flaeche 3 polygon FENSTER gruen  0.1 3.2,  0.6 3.2,  0.6 4.1,  0.1 4.1
            polygon FENSTER gruen -0.6 3.2, -0.1 3.2, -0.1 4.1, -0.6 4.1
     flaeche 4 polygon FENSTER gruen  0.1 3.2,  0.6 3.2,  0.6 4.1,  0.1 4.1
            polygon FENSTER gruen -0.6 3.2, -0.1 3.2, -0.1 4.1, -0.6 4.1
      translation 1 -1.5 0
   RechterTurm gelb wie LinkerTurm translation 0 3 0
   Zwischdach grau translation voll
      basis 0 -0.5 0, 0 0.5 0, 0 0.5 6, 0 0 6.5, 0 -0.5 6
      schub_vektor 2 0 0
} translation x -2
```

```
gruppe MITTELTEIL {
   Hauptschiff grau translation voll
      basis 0 -2 0, 0 2 0, 0 2 4.5, 0 0 6.5, 0 -2 4.5
      schub_vektor x 7
      muster
        flaeche 2 polygon FENSTER gruen 0.2 -1.3, 0.7 -1.3, 0.7 1.3, 0.2 1.3
                  polygon FENSTER gruen 1.3 -1.3, 1.8 -1.3, 1.8 1.3, 1.3 1.3
                  polygon FENSTER gruen -2.2 -1.3, -1.7 -1.3, -1.7 1.3, -2.2 1.3
                  polygon FENSTER gruen -1.1 -1.3, -0.6 -1.3, -0.6 1.3, -1.1 1.3
        flaeche 5 polygon FENSTER gruen 0.6 -1.3, 1.1 -1.3, 1.1 1.3, 0.6 1.3
                  polygon FENSTER gruen 1.7 -1.3, 2.2 -1.3, 2.2 1.3, 1.7 1.3
                  polygon FENSTER gruen -1.8 -1.3, -1.3 -1.3, -1.3 1.3, -1.8 1.3
                  polygon FENSTER gruen -0.7 -1.3, -0.2 -1.3, -0.2 1.3, -0.7 1.3
   Stuetze1 grau translation voll
      basis 0 0 0, 0 0.25 0, 0 0 2 schub_vektor x 0.5
      translation 1 2 0
   Stuetze2 grau wie Stuetze1 translation x 2.45
   Stuetze3 grau wie Stuetze2 translation x 2.45
   Stuetze4 grau translation voll
      basis 0 -0.25 0, 0 0 0, 0 0 2 schub_vektor x 0.5
      translation 1 -2 0
   Stuetze5 grau wie Stuetze4 translation x 2.45
   Stuetze6 grau wie Stuetze5 translation x 2.45
   Sakristei gelb drehflaeche voll
      drehwinkel 180
      meridian 0 -1.5 0, 0 -1.5 3, 0 0 4.5 rot_zahl 9
      translation x 7
   Tuermchen gelb drehflaeche voll
      meridian 0.5 0.5 0, 0.5 0.5 1.5, 0 0 2.5   rot_zahl 4
      muster
        flaeche 1 polygon FENSTER gruen -0.2 -0.2, 0.2 -0.2, 0.2 0.3, -0.2 0.3
        flaeche 2 polygon FENSTER gruen -0.2 -0.2, 0.2 -0.2, 0.2 0.3, -0.2 0.3
        flaeche 3 polygon FENSTER gruen -0.2 -0.2, 0.2 -0.2, 0.2 0.3, -0.2 0.3
        flaeche 4 polygon FENSTER gruen -0.2 -0.2, 0.2 -0.2, 0.2 0.3, -0.2 0.3
      translation 5.5 0 6.5
   Turmaufsatz1 gelb translation voll
      basis 0 0 0.5, 0 0.5 0, 0 0.5 0.5   schub_vektor x 1
      translation 5 0 6
   Turmaufsatz2 gelb translation voll
      basis 0 -0.5 0, 0 0 0.5, 0 -0.5 0.5   schub_vektor x 1
  translation 5 0 6
}
gruppe BASIS {
   WIESE gruen polygon normal z bezugspunkt 0 0 0
   punkte 10 -5, 10 5, -5 5, -5 -5
}
```

Listing zu Farbbild 11:

```
TISCHLAMPE

augpunkt 15 25 10
licht 150 50 250
spiegel quadrat normal z bezugspunkt -8 -8 -7 seite 16
vergroesserung 1.4

10 bausteine
```

```
gruppe UNTERER_TEIL {
/*
/* Statt dem Spiegel kann dieser Tisch benutzt werden */
   TISCHPLATTE grau quader voll 12 12 1 translation -6 -6 -2
*/
   LAMPENNASIS gelb drehflaeche voll
      meridian 4 0 -1, 3.5 0 0   rot_zahl 16
   UNTERER_BUEGEL grau translation voll
      basis 0 0 0, 1.2 0 0, 1.2 1.2 0, 0 1.2 0
      schub_vektor -4 0 6
      translation -1.6 -0.6 0
   KNOPFHALTER gelb drehflaeche voll
      meridian 0.6 0 0, 0.6 0 0.5 rot_zahl 8 translation x 1.5
   KNOPF gruen drehflaeche voll
      meridian 0.3 0 0.5, 0.3 0 1 rot_zahl 8 translation x 1.5
} translation z -6
gruppe OBERER_TEIL {
   MITTLERER_BUEGEL grau translation voll
      basis 0 0 0, 1.2 0 0, 1.2 1.2 0, 0 1.2 0
      schub_vektor 0 0 4
      translation -5.6 -0.6 0
   OBERER_BUEGEL grau allgemein voll
      vertices 0 0 0, 1.2 0 0, 1.2 1.2 0, 0 1.2 0,
               2.2 0 3.8, 3.0 0 3, 3.0 1.2 3, 2.2 1.2 3.8
      kanten   0 1, 1 2, 2 3, 3 0, 0 4, 1 5, 2 6, 3 7, 4 5, 5 6, 6 7, 7 4
      flaechen 0 1 5 4 , 1 2 6 5 , 2 3 7 6 , 3 0 4 7
      translation -5.6 -0.6 4
   VERBINDUNGSSTUECK grau allgemein voll
      punkte  2.2 0 3.8, 3.0 0 3,   3.0 1.2 3,   2.2 1.2 3.8,
               3.8 0.2 4.5, 4.1 0.2 3.6, 4.1 1.0 3.6, 3.8 1.0 4.5
      kanten    0 1, 1 2, 2 3, 3 0, 0 4, 1 5, 2 6, 3 7, 4 5, 5 6, 6 7, 7 4
      flaechen  0 1 5 4 , 1 2 6 5 , 2 3 7 6 , 3 0 4 7
      translation -5.6 -0.6 4
   LAMPENSCHIRM_OBEN gruen drehflaeche voll
      meridian 1.5 0 0, 1.5 0 2, 0.5 0 2.2 rot_zahl 12
      rotation y -20 translation 0 0 8
   LAMPENSCHIRM_UNTEN gruen drehflaeche hohl
      meridian 5.5 0 -5, 1.5 0 0 rot_zahl 12
      +deckel
      rotation y -20 translation 0 0 8
}
```

Listing zu Farbbild 12:

```
MIKROMETER

punkte 1100
augpunkt 25 -20 30
licht -140 -75 100
vergroesserung 1.15
spiegel rechteck normal z bezugspunkt -14 -8 -2.8 seiten 23 16
fixiere 0 0 -2.6

30 bausteine
```

```
gruppe BLEISTIFT1 {
   HOLZ gelb drehflaeche voll
      meridian 0 0.5 -5, 0 0.5 5, 0 0.3 5.6  rot_zahl 6
   GRAPHIT schwarz drehflaeche voll
      meridian 0.3 0 5.6, 0 0 6.5  rot_zahl 6
} rotation y 90  rotation z -20 translation -6 -6 -2.2

gruppe BLEISTIFT2 wie BLEISTIFT1 rotation z 15 translation y 1.5

gruppe HAUPT_ZYLINDER {
   stoppel
      gelb drehflaeche voll
      meridian 1 0 11, 1 0 12
      rot_zahl 16
   drehzyl_plus_kegelstumpf
      grau drehflaeche voll
      meridian 1.2 0 4.1, 1.7 0 5.4, 1.7 0 11
      rot_zahl 16
   schmalstueck
      grau drehflaeche voll
      meridian 1.2 0 2.9, 1.2 0 4.1
      rot_zahl 16
   scheibe1
      grau drehflaeche voll
      meridian 1.7 0.33 1.7, 1.7 0.33 2.9
      rot_zahl 16
   grosse_scheibe
      gelb drehflaeche voll
      meridian 2.6 0.5 0.9, 2.6 0.5 1.7
      rot_zahl 16
   scheibe2
      grau drehflaeche voll
      meridian 1.7 0.33 0, 1.7 0.33 0.9
      rot_zahl 16
   grosser_zapfen
      grau drehflaeche voll
      meridian 0.9 0 -2.8, 0.9 0 0
      rot_zahl 16
   kleiner_zapfen
      gruen drehflaeche voll
      meridian 0.5 0 -3.8, 0.5 0 -2.8
      rot_zahl 16
   durchdringung1 gelb box voll 0.8 0.9 0.6
      translation -0.9 1.7 -0.3
      rotation y 90
   durchdringung2 gelb box voll 0.6 0.9 0.6
      translation -2.4 1.7 -0.3
      rotation y 90
} rotation y -90 rotation x 20  rotation y -6 rotation z 15

gruppe VORDERTEIL {
   gegen_zapfen
      gruen drehflaeche voll
      meridian 0.9 0 -5.3, 0.9 0 -4.7
      rot_zahl 16
      rotation y -90
```

```
   rand_zylinder
      grau drehflaeche voll
      meridian 1.3 0.3 -8.2, 1.3 0.3 -5.3
      rot_zahl 16
      rotation y -90
   vordere_box gelb box voll 2.3 1.3 0.6
      translation 5.7 1.3 -0.3
} rotation x 20  rotation y -6 rotation z 15

gruppe BUEGEL {
   buegel1
      gelb translation voll
      base_points 5.2 0 -0.3, 5.023 1.346 -0.3, 2.801 0.7506 -0.3, 2.9 0 -0.3
      trans_vector z 0.6
   b2 gelb wie buegel1 rotation z -15
   b3 gelb wie buegel1 rotation z -30
   b4 gelb wie buegel1 rotation z -45
   b5 gelb wie buegel1 rotation z -60
   b6 gelb wie buegel1 rotation z -75
   b7 gelb wie buegel1 rotation z -90
   b8 gelb wie buegel1 rotation z -105
   b9 gelb wie buegel1 rotation z -120
   b10 gelb wie buegel1 rotation z -135
   b11 gelb wie buegel1 rotation z -150
   b12 gelb wie buegel1 rotation z -165
} translation 2.8 2.6 0 rotation x 20  rotation y -6 rotation z 15
```

Listing zu Farbbild 13:

```
PHOTOKAMERA

augpunkt 10 20 20
licht 150 -50 100
vergroesserung 1.3
fixiere zentrum

13 bausteine

gruppe GEHAEUSE {
   zentral-koerper  grau translation voll
      basis 1.5 -5.6 0, 1.5 4.4 0, 1 4.9 0, -1.5 4.9 0, -2 4.4 0,
            -2 -5.6 0,  -1.5 -6.1 0, 1 -6.1 0
      schub_vektor z 6
   aufziehhebel_1 gelb translation voll
      basis 0 0 0, 1.5 0 0, 1.5 0.5 0, 0.5 1.3 0, 0 1.3 0
      schub_vektor z 0.5
      translation -1 -5.7 6.15
   aufziehhebel_2 gelb quader voll 0.5 1 0.5
      translation -1 -4.4 6.15
   blitzaufstecker grau allgemein voll
      punkte   0 0 0, 3.5 0 0, 3.5 5 0, 0 5 0,
               1 2 2.5, 3 2 2.5, 3 3 2.5, 1 3 2.5
      kanten   0 1, 2 3, 0 4, 1 5, 2 6, 3 7, 4 5, 5 6, 6 7, 7 4
      flaechen 0 1 5 4 , 1 2 6 5 , 2 3 7 6 , 3 0 4 7 , 4 5 6 7
      translation -2 -2.5 6
   ausloeser gelb quader voll 1 1 0.5
      translation -0.5 3 6
```

```
} translation z -3

gruppe OBJEKTIV {
   aufsatz grau drehflaeche voll
      meridian 2.2 0 1.5, 2.2 0 1.7 rot_zahl 16
   objektiv_1 grau drehflaeche voll
      meridian 2.0 0 1.7, 2.0 0 2.5  rot_zahl 16
   blenden_ring gelb drehflaeche voll
      meridian 2.2 0 2.5, 2.2 0 3.5 rot_zahl 16
   objektiv2 grau drehflaeche voll
      meridian 1.75 0 3.5, 1.75 0 4.5 rot_zahl 16
   entfernungs_ring gelb drehflaeche voll
      meridian 2 0 4.5, 2 0 5.5 rot_zahl 16
   deckel gruen drehflaeche voll
      meridian 1.75 0 5.5, 1.75 0 6 rot_zahl 16
      muster flaeche 18 reg. 16-Eck GLAS schwarz mitte 0 0 eckpunkt 1.5 0
} rotation y 90

gruppe MOTOR {
   motor_gehaeuse grau allgemein voll
      punkte   0 0 0, 0.7 0.7 0, 0.7 2.6 0, 0 3.3 0,
               0 0 6, 0.7 0.7 5, 0.7 2.6 5, 0 3.3 5, 0 2.6 6
      kanten   0 1, 1 2, 2 3, 1 5, 2 6, 3 7, 6 8, 7 8, 4 5, 5 6, 6 7
      flaechen 0 3 2 1 , 0 1 5 4 , 1 2 6 5 , 2 3 7 6 , 4 5 6 8 , 6 7 8
} translation 1.5 -5.6 -3

gruppe KNOEPFE {
   selbstausloeser gelb quader voll 0.25 1.4 0.5
} translation 1.5 2.5 -1.5
```

Listing zu Farbbild 14:

```
FUSSBALL-POKAL

augpunkt -50 25 27
licht -65 -20 70
fixiere zentrum
spiegel quadrat normal z bezugspunkt -15 -15 -14.3 seite 30
vergroesserung 1.2

2 bausteine

STAENDER gelb drehflaeche voll
   meridian -2.73 -3.75 -14, -1.82 -2.5 -8.45 rot_zahl 5

FUSSBALL grau drehflaeche voll
   meridian 0 -3.09 8.45,  0 -6.18 6.54,  -2.94 -7.13 4.63,  2.94 -7.13 4.63,
   -1.82 -8.68 1.54,  1.82 -8.68 1.54,  -3.63 -8.09 -1.54,  3.63 -8.09 -1.54,
   -1.82 -7.5 -4.63,  1.82 -7.5 -4.63,  -3.63 -5 -6.54,  -1.82 -2.5 -8.45
   rot_zahl 5
   kanten    verbinde 1 mit -1, 2
             verbinde 2 mit 3, 4
             verbinde 3 mit 5
             verbinde 4 mit -3, 6
             verbinde 5 mit 6, 7
             verbinde 6 mit 8
             verbinde 7 mit 9
```

```
            verbinde 8 mit 10, -7
            verbinde 9 mit 10, 11
            verbinde 10 mit -11
            verbinde 11 mit 12
            verbinde 12 mit -12
  flaechen
     6 prototypen
     6 punkte 12 -12 -11 10 9 11
     6 punkte 7 9 10 8 6 5
     6 punkte 4 -3 -2 -1 1 2
     6 punkte 6 8 -7 -5 -3 4
     5 punkte -11 -9 -7 8 10
     5 punkte 2 3 5 6 4
     +deckel +boden
  muster flaeche 21 anmalen 5-ECK gruen
         flaeche 22 anmalen 5-ECK gruen
         flaeche 23 anmalen 5-ECK gruen
         flaeche 24 anmalen 5-ECK gruen
         flaeche 25 anmalen 5-ECK gruen
         flaeche 26 anmalen 5-ECK gruen
         flaeche 27 anmalen 5-ECK gruen
         flaeche 28 anmalen 5-ECK gruen
         flaeche 29 anmalen 5-ECK gruen
         flaeche 30 anmalen 5-ECK gruen
         flaeche deckel anmalen 5-ECK gruen
         flaeche boden anmalen 5-ECK gruen
```

Listing zu Farbbild 15:

```
POKERWUERFEL

punkte 3000      /* bei Speichermangel auf 2000 reduzieren und den vierten
                    und fuenften Wuerfel auskommentieren */
augpunkt 45 41 29
licht -50 150 120
vergroesserung 0.75
spiegel quadrat normal z bezugspunkt -18 -18 -4 seite 36
fixiere 0 0 -3

13 bausteine

/*
/* Wenn Sie auf dem Tisch und nicht auf einem Spiegel pokern wollen
   einfach die Kommentare entfernen und die Zeile spiegel oben löschen */
BASISEBENE grau allgemein hohl
   punkte -18 -18 -4 , -18 18 -4 , 18 18 -4 , 18 -18 -4
   kanten 0 1 , 1 2, 2 3, 3 0
   flaechen 0 1 2 3
*/
gruppe WUERFEL {
   1.WUERFEL gelb allgemein voll
      punkte 4  4 3 , 4  3 4 , 3  4 4 ,   -4  4 3 , -3  4 4 , -4  3 4 ,
             4 -4 3 , 4 -3 4 , 3 -4 4 ,   -4 -4 3 , -3 -4 4 , -4 -3 4 ,
             4  4 -3 , 4  3 -4 , 3  4 -4 ,   -4  4 -3 , -3  4 -4 , -4  3 -4 ,
             4 -4 -3 , 4 -3 -4 , 3 -4 -4 ,   -4 -4 -3 , -3 -4 -4 , -4 -3 -4
```

```
    kanten 0 1 , 0 2 , 1 2 , 3 4 , 3 5 , 4 5 ,
           6 7 , 6 8 , 7 8 , 9 10 , 9 11 , 10 11 ,
           12 13 , 12 14 , 13 14 , 15 16 , 15 17 , 16 17 ,
           18 19 , 18 20 , 19 20 , 21 22 , 21 23 , 22 23 ,
           1 7 , 2 4 , 5 11 , 8 10 ,
           0 12 , 3 15 , 9 21 , 6 18 ,
           14 16 , 17 23 , 22 20 , 19 13
    flaechen 0 12 14 16 15 3 4 2 , 5 11 9 21 23 17 15 3 ,
             8 6 18 20 22 21 9 10 , 6 18 19 13 12 0 1 7 ,
             13 14 16 17 23 22 20 19 , 1 2 4 5 11 10 8 7 ,
             0 2 1 , 3 5 4 , 9 11 10 , 6 7 8 ,
             12 13 14 , 15 16 17 , 21 22 23 , 18 19 20
    muster
       flaeche 5 reg. 12-Eck KREIS schwarz mitte  0  0 eckpunkt  1  0
       flaeche 2 reg. 12-Eck KREIS schwarz mitte -2 -2 eckpunkt -1 -2
                 reg. 12-Eck KREIS schwarz mitte  2  2 eckpunkt  3  2
       flaeche 1 reg. 12-Eck KREIS schwarz mitte -2 -2 eckpunkt -1 -2
                 reg. 12-Eck KREIS schwarz mitte  0  0 eckpunkt  1  0
                 reg. 12-Eck KREIS schwarz mitte  2  2 eckpunkt  3  2
       flaeche 3 reg. 12-Eck KREIS schwarz mitte -2 -2 eckpunkt -1 -2
                 reg. 12-Eck KREIS schwarz mitte -2  2 eckpunkt -1  2
                 reg. 12-Eck KREIS schwarz mitte  2  2 eckpunkt  3  2
                 reg. 12-Eck KREIS schwarz mitte  2 -2 eckpunkt  3 -2
       flaeche 4 reg. 12-Eck KREIS schwarz mitte -2 -2 eckpunkt -1 -2
                 reg. 12-Eck KREIS schwarz mitte -2  2 eckpunkt -1  2
                 reg. 12-Eck KREIS schwarz mitte  0  0 eckpunkt  1  0
                 reg. 12-Eck KREIS schwarz mitte  2  2 eckpunkt  3  2
                 reg. 12-Eck KREIS schwarz mitte  2 -2 eckpunkt  3 -2
       flaeche 6 reg. 12-Eck KREIS schwarz mitte -2 -2 eckpunkt -1 -2
                 reg. 12-Eck KREIS schwarz mitte -2  2 eckpunkt -1  2
                 reg. 12-Eck KREIS schwarz mitte  0  2 eckpunkt  1  2
                 reg. 12-Eck KREIS schwarz mitte  0 -2 eckpunkt  1 -2
                 reg. 12-Eck KREIS schwarz mitte  2  2 eckpunkt  3  2
                 reg. 12-Eck KREIS schwarz mitte  2 -2 eckpunkt  3 -2
  2.WUERFEL gruen wie 1.WUERFEL
    rotation x 90  rotation y 90  rotation z 30
    translation x 12
  3.WUERFEL grau wie 1.WUERFEL
    rotation  euler 110 30 130  /* irgendwelche Euler-Winkel */
    translation 1 10.5 1.5
  4.WUERFEL gelb wie 1.WUERFEL
    rotation y 180
    rotation z 60
    translation 1 -10 0
  5.WUERFEL gruen wie 1.WUERFEL
    rotation x 90
    rotation z 30
    translation -10 -11 0
}
```

Listing zu Farbbild 16:

```
SCHRAUBSTOCK

augpunkt 15 20 12
licht 50 250 300
vergroesserung 1.0
spiegel rechteck normal y bezugspunkt 3 -4.5 -5 seiten 18 12
fixiere -4 -2 0

30 bausteine

gruppe SPANN_SCHLUESSEL {
   SCHEIBE1 gelb drehflaeche voll
      meridian 0.5 0 2.0, 0.5 0 2.3, 0.3 0 2.5 rot_zahl 16
   TEIL1 grau drehflaeche voll
      meridian 0.25 0 1.5, 0.25 0 2.0 rot_zahl 16
   SPINDEL gelb drehflaeche voll drehwinkel 300
      meridian 0.8 0 0, 0.8 0 1.5 rot_zahl 16 rotation z 150
   TEIL2 grau drehflaeche voll
      meridian 0.25 0 -3.3, 0.25 0 0 rot_zahl 16
   SCHEIBE2 gelb wie SCHEIBE1 rotation x 180 translation z -1.3
   ACHSE1 grau drehflaeche voll
      meridian 0.4 0 0, 0.4 0 0.5 rot_zahl 16
      rotation y -90  translation -0.7 0 0.75
} rotation achsenpunkt 0 0 0.75 achsenrichtung x winkel -45
  translation -0.3 0 -0.25

gruppe BLOCK1 {
   OBERTEIL grau quader voll 2 4 2.5
   /* Das folgende Verbindungsstueck ist eine Art Trichter.
      Die Seitenflaechen 0 1 5 4 und 2 3 7 6 sind nicht eben und muessen
      daher jeweils in 2 Dreiecke zerlegt werden:
      0 1 5 4 ==> 1 0 4 + 1 4 5  (zusaetzliche Kante 1 4) bzw.
      2 3 7 6 ==> 2 3 7 + 2 7 6  (zusaetzliche Kante 2 7)
   */
   VERENGUNG grau allgemein voll
      punkte   0.05 0.5 -0.4, 1.7 0.5 -1.1, 1.7 3.5 -1.1,  0.05 3.5 -0.4,
                                /* Einmuendung in das gebogene Stueck */
               0 0 0, 2 0 0, 2 4 0, 0 4 0  /* Einmuendung in den Quader */
      kanten   /* 0 1, 1 2, 2 3, 3 0        Basisrechteck, nicht notwendig */
               /* 4 5, 5 6, 6 7, 7 4, 1 4   Deckrechteck, nicht notwendig */
               0 4, 1 5, 2 6, 3 7,  /* Notwendige Kanten */
               1 4, 2 7             /* zusaetzliche Kanten */
      flaechen /* 0 1 2 3 , 4 5 6 7 Basis- u. Deckrechteck nicht notwendig */
               1 0 4 , 1 4 5 , 1 2 6 5 , 2 3 7 , 2 7 6 , 3 0 4 7
} translation -3.5 -2 0

gruppe BACKE1 {
   AUFSATZ1 gruen quader voll 0.5 5 1 translation -2.5 0 0.5
   BACKE1 gelb translation voll
      basis 0 0 0,  -2 0 0, -2 0 1.5, -1 0 1.5 schub_vektor y 5
} translation -1.5 -2.5 2.5
```

```
gruppe BACKE2 {
   AUFSATZ2 gruen wie AUFSATZ1 translation x 0.5
   BACKE2 gelb wie BACKE1 rotation z 180 translation x -7.0
   AUFSATZ grau translation voll
      basis 0 -3.5 0, 0 3.5 0, 0 2.5 1.5, 0 -2.5 1.5 schub_vektor x 2
      translation -5.5 0 1
   WAGEN grau quader voll 3 4 1.5 translation -8.5 -2 1
   TRAPEZ_FUEHRUNG1 grau translation voll
      basis 0 -1.5 0, 0 1.5 0, 0 1.2 1.5, 0 -1.2 1.5 schub_vektor x 1
      translation -4.5 0 -0.5
   EINSATZ gruen drehflaeche voll
      meridian 0.5 0 0, 0.5 0 0.5, 0.4 0 0.6 rot_zahl 16
      translation -7 0 2.5
} translation x -1.5

gruppe HALTERUNG {
   SENKRECHT gelb quader voll 2 7 5  translation y -3.5
   WAAGRECHT1 gelb quader voll 5 1 2.5 translation -5 -3 2.5
   WAAGRECHT2 gelb wie WAAGRECHT1 translation y 5
   TRAPEZ_FUEHRUNG2 grau translation voll
      basis 0 -1.5 0, 0 1.5 0, 0 1.2 1.5, 0 -1.2 1.5 schub_vektor x 3
      translation -3 0 3.5
   HINT_ABSCHLUSS gelb quader voll 1 4 2.5 translation  -5 -2 2.5
   BODEN gruen quader voll 4 4 1 translation -5 -2 2.5
} translation -8 0 -4

gruppe ACHSE {
   ACHSE gruen drehflaeche voll
      meridian 0.5 0 0.5, 0.5 0 2 rot_zahl 16
      rotation y -90 translation -3 0 0.5
}

gruppe BUEGEL {
   B1 grau translation voll
      basis 0 0 -4.5, 0 0 -2.75, 0.478 0 -2.71, 0.78 0 -4.43
      schub_vektor y 3
   B2 grau wie B1 rotation y -10
   B3 grau wie B1 rotation y -20
   B4 grau wie B1 rotation y -30
   B5 grau wie B1 rotation y -40
   B6 grau wie B1 rotation y -50
   B7 grau wie B1 rotation y -60
} translation -6 -1.5 0.5
```

Listing zu Farbbild 17:

```
"SPINNE"

augpunkt 10 10 10
licht 50 100 150

1 baustein
```

```
GRAPH_MIT_EIGENER_FORMEL
   gruen mathematisch concave
   funktion formel (sin(x)*((x*x)+(y*y)-1.5))
      x-werte  -1.5 1.5
      y-werte  -1.5 1.5
      schnitte_parallel_x 18
      schnitte_parallel_y 18
      schichtenlinien 15
```

Listing zu Farbbild 18:

```
TORUS_auf_Spiegel

augpunkt 20 20 10
licht 50 90 100
spiegel quadrat normal z bezugspunkt -7 -7 -3 seite 14

1 baustein

3/4torus
   gelb mathematisch konkav
   torus
      rad_des_mittenkreises  4
      rad_des_rot_kreises    2
      anfangswinkel_am_meridian 90
      endwinkel_am_meridian     450
      anfangswinkel_am_mittenkreis 90
      endwinkel_am_mittenkreis    360
      parallelkreise 20
      meridiankreise 20
      schachbrett gruen
```

Listing zu Farbbild 19:

```
WEINGLAESER_AM_TISCH

punkte 2000
augpunkt 20 20 10
licht 50 90 100
fixiere zentrum

spiegel rechteck normal z bezugspunkt -10 -6 0 seiten 20 12

6 bausteine

gruppe 1 {
WEINGLAS1
   gelb mathematisch konkav drehflaeche
   meridian mer_groesse 15 rot_zahl 15
   1.4 0 -3, 0.3 0 -2.3, 0.3 0 -1.3, 0.3 0 -0.8, 0.7 0 -0.5,
   0.3 0 -0.3, 0.3 0 0.2, 1.6 0 1, 1.9 0 2, 1.5 0 3, 1.6 0 4
   translation -6 -3 0
   WEINGLAS2 gelb wie WEINGLAS1 translation x 12
   WEINGLAS3 gelb wie WEINGLAS1 translation y 6
   WEINGLAS4 gelb wie WEINGLAS3 translation x 12
} translation z 3
```

Listing zu Farbbild 20:

```
7_WUERFEL

augpunkt -60 30 30
licht -150 80 250
vergroesserung 1.1
fixiere 15 0 0
spiegel quadrat normal x bezugspunkt 20 -20 -20 seite 40


13 bausteine

gruppe CUBE_COMPLEX {
   MITTENWUERFEL gelb quader voll 8 8 8
      muster flaeche deckel reg. 3-Eck KREIS schwarz mitte 0 0 eckpunkt 3 0
      translation -4 -4 -4
   QUADER3 gruen quader voll 2 2 7 translation -1 -1 -11 rotation y 90
   QUADER4 gruen wie QUADER3 rotation y 180
   QUADER5 gruen wie QUADER3 rotation z 90
   QUADER6 gruen wie QUADER3 rotation z -90
   LINKER-WUERFEL   grau   wie MITTENWUERFEL translation y -15
   RECHTER-WUERFEL  grau   wie MITTENWUERFEL translation y  15
   VORDERER-WUERFEL  gruen  wie MITTENWUERFEL translation  x 15
^   HINTERER-WUERFEL   gruen  quader voll 8 8 8
      muster
         flaeche deckel polygon QUADRAT gelb -2 -2 , -2 2 , 2 2 , 2 -2
      translation -19 -4 -4
}

gruppe OBERE_WUERFEL {
   OBERER-WUERFEL gelb wie MITTENWUERFEL
   QUADER1 gruen quader voll 2 2 7 translation -1 -1 -11
} translation z 15

gruppe UNTERE_WUERFEL {
   UNTERER-WUERFEL gelb wie MITTENWUERFEL
   QUADER2 gruen wie QUADER1
} translation z -15
```

Die restlichen Listings befinden sich entweder auf der Programm- oder der Datendiskette.

Anhang B

Die Tastaturbelegung der Benutzerschnittstelle

Hier eine Zusammenfassung der Tastaturbelegung der grafischen Benutzeroberfläche in alphabetischer Reihenfolge:

Tastaturabkürzungen der Menü-Punkte:

Taste	Menü-Punkt
[e]	Datei->Arbeitsumgebung...->Text-Editor
[l]	Datei->Laden...->Datenfile
[s]	Datei->Speichern als...->Datenfile
[RAmiga]+[A]	Animation->Start...->Autorotation
[RAmiga]+[D]	Edit->Datenfile
[RAmiga]+[E]	Edit->Augpunkt (Eye)
[RAmiga]+[F]	Edit->Animations-Skript
[RAmiga]+[I]	Animation->Start...->Interaktiv
[RAmiga]+[N]	Anzeigen->Nachricht
[RAmiga]+[Q]	Datei->Ende
[RAmiga]+[R]	Animation->Start...->Video Replay
[RAmiga]+[S]	Animation->Start...->Animations-Skript
[RAmiga]+[T]	Edit->Verdreh-Winkel (Twist)
[RAmiga]+[V]	Animation->Start...->Autorot -> Video
[RAmiga]+[W]	Animation->Start...->Skript -> Video
[RAmiga]+[1]	Edit->Lichter...->1
[RAmiga]+[2]	Edit->Lichter...->2
[Shift]+[L]	Datei->Laden...->Animations-Skript
[Shift]+[S]	Datei->Speichern als...->Animations-Skript

Tastaturabkürzungen in den Kommunikationsfenstern:

[1]	erniedrigt den Wert des aktiven Zahlen-Symbols um 100
[2]	erniedrigt den Wert des aktiven Zahlen-Symbols um 10
[3]	erniedrigt den Wert des aktiven Zahlen-Symbols um 1
[4]	erklärt das linke Zahlensymbol zum aktiven
[5]	erklärt das mittlere Zahlensymbol zum aktiven
[6]	erklärt das rechte Zahlensymbol zum aktiven
[7]	erhöht den Wert des aktiven Zahlen-Symbols um 100
[8]	erhöht den Wert des aktiven Zahlen-Symbols um 10
[9]	erhöht den Wert des aktiven Zahlen-Symbols um 1
[Return]	wirkt wie das O.K.-Symbol
[Esc]	wirkt wie das Abbruch-Symbol
[u]	wirkt wie das Undo-Symbol
[a]	aktiviert das aktive Symbol

Anhang C

Die Tastaturbelegung des Animators

Egal mit welchen Optionen (Autorotation, Animations-Skripten, Video ...) der Animator gestartet wurde, die Szene kann jederzeit interaktiv über die Tastatur verändert werden.

Hier noch einmal eine Zusammenfassung aller Tasten und ihrer Funktionen:

Menüsteuerung aufrufen:

[']: Die Taste über der Tabulatortaste (am linken Rand der Tastatur) aktiviert die Menüsteuerung.

Beenden der Animation:

[q] bzw. [Esc]: Die Animation wird beendet. Wenn Sie diese Taste irrtümlich gedrückt haben und rechtzeitig sofort darauf [Shift]+ [q] drücken, wird der Ausstiegsbefehl ignoriert.

Änderung des Zeichenmodus:

[w] (= Wireframe) schaltet den Drahtgittermodell-Modus ein

[h] (= Hide) schaltet den Plottermodus ein (alle verdecketen Linien werden entfernt)

[s] (= Shades) schaltet die Schattierung der Objekte ein

[c] (= Cast Shadows) schaltet die Schlagschatten ein

[Shift]+ [C] schaltet die Schlagschatten wieder aus (wirkt wie [s])

Ein- und Ausschalten der Auto-Animation:

[f] schaltet die Auto-Animation aus (freeze = einfrieren)

[Shift]+ [F] schaltet die Auto-Animation ein

Rotationen:

[a] und [Shift]+ [A] verändert den Azimutalwinkel

[e] und [Shift]+ [E] verändert den Höhenwinkel (elevation)

[t] und [Shift]+ [T] verändert den Drehwinkel (twist)

[<] dreht die Szene um 45° um den Azimutalwinkel

[>] dreht die Szene um −45° um den Azimutalwinkel

[^] dreht die Szene um 45° um den Höhenwinkel

[#] dreht die Szene um −45° um den Höhenwinkel

Spezielle Ansichten:

[1] stellt die Szene von oben gesehen dar (Grundriß)

[2] stellt die Szene von vorne gesehen dar (Aufriß)

[3] stellt die Szene von rechts gesehen dar (Kreuzriß)

[4] stellt die Szene von unten gesehen dar

[5] stellt die Szene von hinten gesehen dar

[6] stellt die Szene von links gesehen dar

[u] (= Undo) kehrt in die Ausgangslage zurück

([1] = vorwärts, [2] = rückwärts)

Verändern der Distanz:

[d] vergrößert die Distanz (Szene erscheint dadurch im Bild kleiner)

[Shift]+ [d] verkleinert die Distanz

[x] vergrößert zweidimensional das Bild der Szene (zoom)

[Shift]+ [x] verkleinert das Bild der Szene

[y] Achsenkreuz und Lichtrichtung zusätzlich anzeigen

[Shift]+ [y] Achsenkreuz und Lichtrichtung nicht mehr anzeigen

Verschieben des Bildes innerhalb des Bildschirms (zweidimensional):

[Num 4]: Nach-links-Bewegen aller Objekte

[Num 6]: Nach-rechts-Bewegen aller Objekte

[Num 8]: Nach-oben-Bewegen aller Objekte

[Num 2]: Nach-unten-Bewegen aller Objekte

[Num 5]: zentriert den Ursprung in der Bildschirmmitte

Schnelle Veränderung bei komplexen Szenen:

[m] schaltet in den Move-Modus, in dem nur noch das Achsenkreuz bewegt wird. Dies ist vor allem nützlich, wenn die Szene schon aus Hunderten oder gar Tausenden Punkten besteht. In diesem Modus können Sie die Achsen wie oben mit [a], [e], [t] bzw. [Shift]+ [a], [Shift]+ [e], [Shift]+ [t] um den Azimutal-, Höhen- und Verdrehungswinkel rotieren. [d] bzw. [Shift]+ [d] bewirkt eine Änderung der Distanz, [x] bzw. [Shift]+ [x] eine Vergrößerung.

Mit [u] (wie undo) werden alle Änderungen zurückgenommen.

Ausstieg aus diesem Modus mit [q] oder [Esc] (nur einmal drücken!)

Verändern der Lichtposition (des Lichtvektors):

[l] schaltet die Darstellung des Achsenkreuzes und des Lichtvektors ein:

Nun kann mit den Tasten [4] und [6] der Azimutalwinkel des Lichts verändert werden, und die Tasten [8] und [2] verändern den Höhenwinkel des Lichts.

Mit [u] (wie undo) werden alle Änderungen zurückgenommen.

Ausstieg aus diesem Modus mit [q] oder [Esc] (nur einmal drücken!)

[g] macht die Szene »geozentrisch« (Lichtquelle starr mit der Szene verbunden)

[Shift]+ [g] macht die Szene wieder »heliozentrisch« (Lichtquelle unabhängig von der Lage der Objekte).

Farbpaletten anzeigen:

[k] zeigt die verwendete Farbpalette

[Shift]+ [K] schaltet sie wieder aus

Konstruktionshilfen:

[b] stellt die »bounding boxes« der Objektgruppen, wenn diese berechnet wurden, dar (nicht im Drahtgittermodell!)

[Shift]+ [B] schaltet wieder in den Normalmodus zurück

[o] zeichnet die Umrisse der einzelnen Körper stark nach (outline)

[Shift]+ [O] hebt [o] auf

Zeichenvorgang:

[z] zeigt den Zeichenvorgang für drei Bilder (das heißt, es wird nicht im Hintergrund gezeichnet, sondern am sichtbaren Bildschirm)

[Shift]+ [Z] veranlaßt das Programm, wieder im Hintergrund zu zeichnen

Screen:

[.] zeigt die Titelleiste des Screens: Ist sie sichtbar, kann der Screen mit der Maus bewegt (heruntergezogen) werden

[:] verdeckt die Titelleiste wieder

[Shift]+ [H] Vorbereitung des Bildschirms für einen Graphic-Dump

Ausgeben der Szene:

[p] plottet die gesamte Szene aus

Abspeichern der Animation als Video:

[v] schaltet den Videomodus ein

[Shift]+ [V] schaltet wieder aus

[i] speichert das sichtbare Bild als IFF-Bild

Anhang D

Schlüsselwörter der Programmiersprache »3D«

Die folgenden (immer klein geschriebenen) Schlüsselwörter werden vom 3D-Sprinter an geeigneter Stelle »verstanden«:

```
+boden              +bottom
+deckel             +top
achsenpunkt         axpoint
achsenrichtung      axdirection
allgemein           general
anfang              start
anmalen             paint
anstieg             inclination
augpunkt            eye
basis               base_points
berge               mountains
bezugspunkt         point_of_reference
bild                frame
bilder              frames
boden               bottom
datenfile           datafile
deckel              top
drahtgitter         wire_frame
drehflaeche         revolution
drehwinkel          rot_angle
durchloechert       perforated
eckpunkt            vertex
ende                end
faktor              factor
fallinien           orthogonals
fixiere             lookat
flaeche             face
flaechen            faces
formel              formula
formel              formula
funktion            function_graph
geozentrisch        geocentric
geschlitzt          sliced
gruppe              group
hohl                hollow
insgesamt           total
kanten              edges
konkav              concave
kugel               sphere
kugel               sphere
licht               light
loxodromen          loxodromes
mathematisch        math_surface
meridian            meridian
```

```
meridian_groesse      meridian_size
mit                   with
mitte                 center
muster                patterns
normal                normal
polygon               poly
prototypen            prototypes
punkte                vertices
quader                box
quadrat               square
radius                radius
raumkurve             curve
rechteck              rectangle
reg.                  regular
rotation              rotation
rot_zahl              order
schachbrett           chess_pattern
schatten              shadows
schattiert            shades
schichtenlinien       intersections
schub_vektor          trans_vector
seite                 length
seiten                lengths
spiegel               mirror
standbild             freeze
strecke               line
streckung             stretch
translation           translation
umriss                silhouette
verbinde              connect
verdeckte_linien      hiddenlines
verdrehung            twist
vergroesserung        enlarge
voll                  solid
welle                 wave
wie                   like
winkel                angle
zentrum               center
zylinder              cylinder
z_links               left_z
z_rechts              right_z
```

Anhang E

Installation des Programms auf der Festplatte

Nichts leichter als das. Ziehen Sie einfach mit der Maus das Piktogramm der 3D-Sprinter-Schublade in eines der Fenster Ihrer Festplatte. Wenn Sie sich das Anklicken des Assign_3D-Sprinter-Programms vor jedem Programmstart ersparen wollen, fügen Sie bitte folgende Zeilen in Ihre Startup-Sequence ein:

```
Assign 3D-Sprinter: Name_der_Festplatte:pfad/3D-Sprinter
Path 3D-Sprinter: Add
```

Wie immer auch hier ein Beispiel:

```
Assign 3D-Sprinter: DH0:Graphik/3D-Sprinter
Path 3D-Sprinter: Add
```

Anhang F

Weiterführende Literatur

Genaueres über den theoretischen Hintergrund und die verwendeten Algorithmen finden Sie in den folgenden Büchern bzw. Fachzeitschriften:

[1] Georg Glaeser, 3D-Programmierung mit BASIC, hpt/Wien, BGT/-Stuttgart, 1986.

[2] Georg Glaeser, Ein schnelles Verfahren zur realistischen Darstellung einfacher Szenen, Informatik-Fachberichte 182, 187–198, Springer/Berlin, 1988.

[3] Georg Glaeser, Problemangepaßte schnelle Algorithmen zur Bestimmung spezieller Flächenkurven, CAD und Computergrafik, Jahrgang 11, Heft 4, Wien 1988.

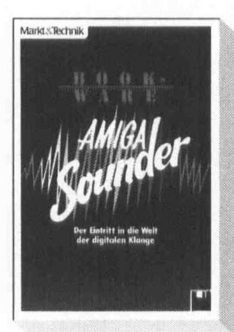
[4] Georg Glaeser, 3D-Graphics in C (Titelvorschlag), erscheint Anfang 1990 bei Springer/New York.

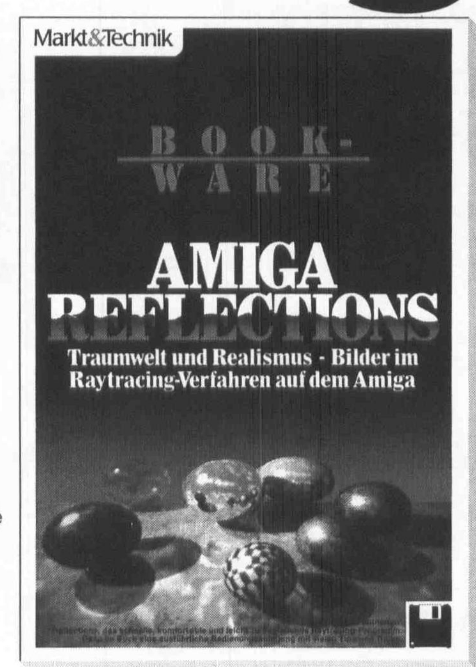


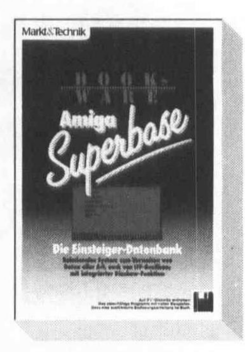
Stichwortverzeichnis

BOOK-
WARE

Amiga







In Vorbereitung:
H. Knappe
Amiga Sounder
Der Amiga Sounder ist ein Komplettpaket für den Einstieg in die Welt der digitalen Klänge. Die beigefügte Platine bildet die Grundlage zum Bau eines 4-Kanal-Sound-Digitizers für wenig Geld.
Lieferbar 3. Quartal 1989,
ca. 120 Seiten,
inkl. 2 Programmdisketten,
inkl. Platine
ISBN 3-89090-709-1
ca. DM 98,-*
(sFr 90,20*/öS 834,-*)

N. Wirsing
Amiga Audio Entwicklerpaket
Dieses Buch macht Sie zum perfekten Amiga-Tontechniker.
Lieferbar 3. Quartal 1989,
ca. 400 Seiten,
inkl. 2 Disketten
ISBN 3-89090-765-2
ca. DM 98,-*
(sFr 90,20*/öS 834,-*)

C. Fuchs
Reflections
Erschließen Sie sich mit Reflections die faszinierenden Möglichkeiten des Raytracings. Mit diesem Programmsystem erzeugen Sie in kurzer Zeit Bilder mit einem auf dem Amiga bisher unbekannten Realitätsgrad.
1989, 156 Seiten,
inkl. Programmdiskette
ISBN 3-89090-727-X
DM 98,-*
(sFr 90,20*/öS 834,-*)

Precision Software
Amiga Superbase
Die Einsteiger-Datenbank mit professionellen Eigenschaften. Leicht erlernbar und bedienbar. Sie brauchen nur noch Ihre Daten einzutippen, alles andere erledigen Sie per Mausklick oder über Pull-down-Menüs - auch den Aufbau einer Datenbank. Das System ist relational, Sie können also verschiedene Datenbanken miteinander verknüpfen. Die Bildschirmdarstellung ist komfortabel: Tabellen oder Formulare mit beliebigen Auswahlkriterien stehen zur Verfügung.
Hardware-Anforderungen:
Amiga 500, 1000 oder 2000 mit mindestens 512 Kbyte Arbeitsspeicher.
1989, 176 Seiten,
inkl. Programmdiskette
ISBN 3-89090-791-1
ca. DM 89,-*
(sFr 81,90*/öS 757,-*)

*Unverbindliche Preisempfehlung



4212/906

Markt&Technik-Produkte erhalten Sie bei Ihrem Buch- oder Computerfachhändler

Amiga 3D-Sprinter

Die Autoren:

Dr. GEORG GLAESER, geboren 1955 in St. Johann im Pongau (Bundesland Salzburg, Österreich). Studium der Mathematik und Geometrie an der Technischen Universität Wien. Seit 1978 Lehrer an der Höheren Technischen Bundeslehranstalt Wien 10, seit 1983 Lektor für Geometrie an der Technischen Universität Wien. 1986/87 Forschungsstipendium für Computergrafik an der Universität Princeton/New Jersey, USA.

THOMAS GROHSER, geboren 1970 in Wien (Österreich), zur Zeit noch Schüler an der Höheren Technischen Bundeslehranstalt Wien 10. Fachrichtung: Elektrotechnik. Neben der Schule entwickelt er seit fünf Jahren Software, die letzten drei Jahre auf dem Amiga.

3D-Sprinter

Der Ruhm des Amiga, der seine steile Karriere im harten Konkurrenzkampf der Computer erklärt, beruht vor allem auf seinen bahnbrechenden Grafikmöglichkeiten und kurzen Rechenzeiten. Doch was wäre er ohne entsprechend brillante Software?

Das Programm 3D-Sprinter ergänzt den Amiga in seinen Eigenschaften und reizt ihn vollständig aus: Grafiken, Schatten, Animationen und Spiegelungen werden, je nach Rechnerkonfiguration, in Echtzeit berechnet.

Die Bedienung ist ebenso einfach wie effizient: Über definierte Objekte kreieren Sie Ihre Grafiken und manipulieren diese dann über die Tastatur, mit der Maus oder mit Hilfe von edierbaren Skript-Dateien. Das Drehen, Vergrößern, Verkleinern, Bewegen etc. ganzer Szenen erfolgt direkt am Bildschirm!

Der 3D-Sprinter ist deshalb ebenso für private Zwecke wie für wissenschaftliche Anwendungen geeignet (z.B. Darstellung von Molekularstrukturen). Er kann nach kurzer Einarbeitungszeit von jedem Computerfreund effektiv eingesetzt werden.

Die Leistungsmerkmale:

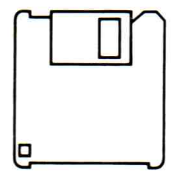
- Einfache Generierung von Objekten: Würfel, Kugeln, Drehkörper, Translationskörper, Funktionsgrafen u.v.a.
- Beschreibung der Objekte wahlweise in Deutsch oder Englisch
- Darstellen der Grafiken als Drahtgittermodell, Plotterzeichnung, schattiert mit und ohne Schlagschatten
- Einzeichnen von Höhenschichten-, Fall- und Schnittlinien bei Funktionsgrafen
- Festlegen der Kamera- und Lichtposition in Animations-Skripten oder durch interaktive Manipulation
- Speichern und Wiederabspielen der Animation
- Alle Auflösungen des Amiga mit bis zu 64 Farben gleichzeitig ansprechbar
- Ausführliches, mit vielen Farbbildern kommentiertes Handbuch mit einer Bedienungsanleitung und einer Einführung in die Grundlagen der 3-D-Grafik: Perspektive, Koordinatensysteme u.v.m.
- Deutsche Programmversion

Hardware-Anforderungen:

Amiga 500/1000/2000/2500 mit 1 Mbyte Arbeitsspeicher (ein eingeschränkter Betrieb ist auch mit 512 Kbyte möglich), Kickstart/Workbench 1.2 (oder höher), ein Diskettenlaufwerk. Empfohlen: zweites Diskettenlaufwerk oder Festplatte, Speichererweiterung auf mindestens 2 Mbyte.

BOOK-WARE
Profi-Software zum Buchpreis

ISBN 3-89090-109-3



4 001057 901094

Markt&Technik

Unverbindliche Preisempfehlung

DM 98,-
sFr 90,20
öS 834,-